OPTIMIZATION IN MULTI-SPINDLE AUTOMATICS —A PROBABILISTIC APPROACH

By

SUMAN DEO KOTHARI

DEPARTMENT OF MECHANICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

SEPTEMBER, 1973.

OPTIMIZATION IN MULTI-SPINDLE AUTOMATICS —A PROBABILISTIC APPROACH

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of
MASTER OF TECHNOLOGY

By
SUMAN DEO KOTHARI

to the
DEPARTMENT OF MECHANICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

SEPTEMBER, 1973.

CERTIFICATE

Certified that this work on 'Optimization in Multi-Spindle Automatics - A Probabilistic Approach' by Suman Deo Kothari has been carried out under my supervision and that this has not been submitted elsewhere for a degree.

Dr. J. L. Batra
Assistant Professor
Department of Mechanical Engineering
Indian Institute of Technology Kanpur

ACKNOWLEDGEMENT

I take this opportunity to place on record my indebtedness to Dr. J. L. Batra for his invaluable help and guidance.

My thanks are due to Messrs R. Rajagopalan and A. R. Lele for their suggestions and fruitful discussions. And I acknowledge the timely help rendered by my friends R. N. Choudhury and N. Ravichandran.

Last but not the least, I am thankful to Mr. J. D. Varma for his excellent typing.

Suman Deo Kothari

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

NOMENCLATURE

C_1	-	Direct labour cost, Rs./min.
C_2	-	Overhead rate, Rs./min.
C_3	-	Machine tool setter rate, Rs./min.
CCH	-	Tool-change cost (for unscheduled tool-replacement), Rs.
CCH_1	-	Tool-change cost of type-a (for scheduled tool-replacement), Rs.
CCH_2	-	Tool-change cost of type-b (for scheduled tool-replacement), Rs.
CG	-	Grinding labour rate, Rs./min.
C_m	-	Total machining and operating cost, Rs./min.
CNP	-	Non-productive time cost, Rs.
CP	-	Profit per component, Rs.
CS	-	Tool-change labour rate, Rs./min.
CT	-	Total cost of production, Rs.
d	-	Depth of cut, inches.
f	-	feed rate, ipr.
f_{max}	-	Maximum feed rate available on the machine tool, ipr.
f_{min}	-	Minimum feed rate available on the machine tool, ipr.
FC	-	Cutting force, lbs.
FC_{max}	-	Maximum cutting force allowed, lbs.
GRT	-	Time required to replace all the cutting tools together, min.
N	-	Number of spindles.
NJ	-	Lot size.

NP - Number of components for which tool-life constraint is to be satisfied.

NS - Number of schedules required to produce the lot.

PC - Horsepower.

PC_{max} - Horse power of a machine (maximum available).

p_1 - Probability level.

p_2 - Probability level.

q - A factor.

S - Cutting speed, fpm.

SI - Tool-replacement interval, min.

$\overline{T}$ - Average system life in unscheduled tool-replacement procedure, min.

TC - Cycle time, min.

TL - Tool-life, min.

TS - Initial machine setting time, min.

t_a - Time required for approach of tools, min.

t_l - Time required for loading and unloading of the job, min.

t_I - Time required for indexing of spindles, min.

V_{max} - Maximum speed, rpm.

V_{min} - Minimum speed, rpm.

α_0, α_1, α_2, and α_3 - Exponent of constant, speed, feed, and depth of cut respectively in force equation.

α_4, α_5, α_6, and α_7 - Exponent of constant, speed, feed and depth of cut respectively in tool-life equation.

σ_{α_0}, σ_{α_1}, σ_{α_2}, and σ_{α_3} - Standard deviations of α_0, α_1, α_2, and α_3 respectively.

σ_{α_4}, σ_{α_5}, σ_{α_6}, and σ_{α_7} - Standard deviation of α_4, α_5, α_6, and α_7 respectively.

η - Power transmission efficiency.

Following notations are for the i^{th} spindle.

C_{ij}	-	A factor for j^{th} tool.
d_{ij}	-	Depth of cut for j^{th} tool, inches.
EC_{ij}	-	Tool edge cost for j^{th} cost, Rs.
f_{ij}	-	Feed rate for j^{th} tool, ipr.
l_{ij}	-	Tool travel length, inches.
L_{ij}	-	Machining length for j^{th} tool, inches.
m_{ij}	-	Number of permissible regrinds for j^{th} tool.
NR_{ij}	-	A factor for j^{th} tool.
n_i	-	Number of operations.
$p_{1_{ij}}$	-	Probability level for tool-life constraint for j^{th} tool.
$p_{2_{ij}}$	-	Probability level for horsepower constraint for j^{th} tool.
p_{2_i}	-	Probability level for horsepower constraint.
$p_{3_{ij}}$	-	Probability level for cutting force constraint.
PC_{ij}	-	Horsepower consumed by j^{th} tool.
T_{ij}	-	Tool-life of j^{th} tool, min.
TG_{ij}	-	Grinding time for j^{th} tool, min.
TP_{ij}	-	Total actual machining time for j^{th} tool, min.
TS_{ij}	-	Tool setting time for j^{th} tool, min.
t_{ij}	-	Tool travel time for j^{th} tool, min.
V_i	-	Cutting speed, rpm.
$\sigma_{T_{ij}}$	-	Standard deviation of tool-life of j^{th} tool.

SYNOPSIS

SUMAN DEO KOTHARI
M. Tech. (Mech.)
Indian Institute of Technology
Kanpur
September, 1973
OPTIMIZATION IN MULTI-SPINDLE AUTOMATICS
- A PROBABILISTIC APPROACH

An analytical methodology is proposed to determine optimum cutting conditions and optimum tool-replacement procedure for manufacturing operation performed on multi-spindle automatics considering tool-life and cutting-force as probabilistic in nature. The optimization is carried out w.r.t. either minimization of cycle time or minimization of total cost of production, considering the various constraints imposed by machine and cutting-tool characteristics, physical process, and the job specifications. The concept of chance constrained programming has been utilized to account for the probabilistic nature of tool-life and cutting-force.

The model consists of a non-linear objective function subject to several non-linear constraints. An interior penalty function approach is employed to help use Sequential Unconstrained Minimization Technique (SUMT). Davidon - Fletcher - Powell Variable Metric Method (DFP) is used to advantage with Golden Search Technique for carrying out unidirectional minimization.

The proposed methodology is computerized to determine optimum machining parameters and to determine the total cost of production for the two tool-replacement procedures. The procedures are : 1) unscheduled tool-replacement and 2) scheduled tool-replacement. It is shown that the optimum cutting conditions, cycle time, and the total cost of production are significantly effected by the probabilistic nature of coefficients in the constraints. The computer package has been tested through an example of a job to be produced on multi-spindle automatic.

In this study the effect of 1) capacity of machine tool and 2) the type of machine tool depending on whether all the spindles have common speed and feed or the spindles having different speeds and feeds has also been investigated.

CHAPTER I

INTRODUCTION

The concept of manufacturing is undergoing a revolution. The traditional thinking that manufacturing is an art is, slowly but surely, being replaced by the idea that it is a scientific branch of engineering. To quote Merchant (33),

> "Basic and long-range technical changes are today challenging the conventional economics of manufacturing. The key weapon to which industry is turning to combat this challenge is new optimization technology; computer-related technology and versatile automation are its main components. The prime technical advance which has made possible extensive use of both of these categories of new technology is the digital computer; manufacturing is apparently already its second largest area of application in industry. Versatile (variable programme) automation, under the impetus of computer's capabilities, is now striving to upgrade its control capabilities from those afforded by feedback and supervisory control to those offered by self - optimizing control. Simultaneously, it is rapidly upgrading its programme input sources from previously restrictive manual and analogue inputs to digital inputs. This process is leading to the ultimate tool for optimization of manufacturing,

direct digital control, in which the control loop is closed through the computer; the fully self - optimizing, fully automated manufacturing system is its potential climax ...."

Discrete item manufacturing has long been receptive to automation innovations. Since 1955, one of the fastest growing areas is the development of Computer Assisted Manufacturing (CAM) systems. The control of digital computer has been fully accepted in the continuous or process industries. However, this acceptance has not been so prevelant in the discrete item industries for certain rather obvious and complex reasons.

An optimizing machine can be thought as a separate section in a fully automated system. The purpose of the machine is to optimize the manufacturing activity in its entirety. The optimizing section may further be divided in order to handle various aspects of manufacturing - raw material control, production planning, scheduling, production, handling of finished products and so forth. The optimizing equipment must be equipped with certain logic, analytical or heuristic, in order to arrive at optimum decisions.

The present work deals with the manufacturing aspect of a concern producing discrete components. The ability of any manufacturing firm to successfully compete in the market is highly dependent on the cost of its manufacturing processes. To quote Cook (9), "... producing satisfactory parts at the lowest

possible cost can be called the 'First Law of Production'.... " In most organizations those decisions which directly relate to the cost of producing a product are made on the basis of experience (often limited) and with the assistance of various charts and tables of "proper operating conditions" put out by the manufacturers of machine tools, cutting-tools, etc. Although the basis for rational selection of manufacturing parameters was set down by Taylor in the early 1900's, only recently people have attempted serious utilization of analytical selection methods.

Considerable work has been done applying analytical and heuristic approaches for single - tool operation using deterministool-life values. It has been recommended that tool - life predictions be made on a probabilistic basis. However, the economics of multi - tool operation with probabilistic tool - life consideration has received very little attention, possibly because of its inherent complexities. It was thus felt that structuring of a model for multi - tool operation and subsequent application of a mathematical programming technique would be a worthwhile effort. Therefore, the present work is directed towards the optimal selection of cutting parameters for multi - spindle automatics when the tool - life is considered probabilistic. Further, the work includes the selection of tool-replacement programme for changing the cutting-tools.

The problem of optimization for multi - spindle automatics is extremely complex due to diversities in the type of operations

performed at each station. This problem is further aggravated due to the probabilistic tool failure characteristics of the tools involved. The problem as such, consists of having a methodology to provide answer for the following decisions :

1. Determination of machining parameters for each station,
2. Determination of optimal tool - replacement policy, and
3. Selection of a machine tool.

To seek a logical answer to the above problem, an analytical model has been formulated. The stochastic nature of constants and exponents in tool - life and cutting - force equation is accounted for by structuring the problem as a chance constrained programming problem. The constraints are imposed due to machine and cutting - tool characteristics and job requirements. The objective functions studied are :

1. Minimization of cycle time, and
2. Minimization of total production cost for a given production lot size.

The model consists of a non - linear objective function subject to non - linear constraints. An interior penalty function approach is employed to help use Sequential Unconstrained Minimization Technique (SUMT). Davidon - Fletcher - Powell Variable Metric (DFP) method is used to advantage with Golden Search Technique for carrying out unidirectional minimization.

The algorithm has been programmed in FORTRAN IV language for use in IBM 7044 digital computers. The computer package is quite general in nature to handle optimization problem for manufacturing operation in which material is removed in form of chips. However, the package is tested only for a job to be produced on multi - spindle automatic machine tool.

The present work is but a step or two leading to the final goal viz. fully automated manufacturing system. However, the eventual realization of such a system will and must be an evolutionary process.

CHAPTER II

LITERATURE SURVEY

The art of metal removing is as old as human civilization itself. New chapters are being added continuously to meet the challenges of time. The state of the art that is witnessed today is constituted of a number of broad areas each of them forming a separate field of its own. The areas relevant to the present work are :

1. Tool - Life and its Characteristics, and
2. Economics of Machining.

The following sections discuss the literature regarding the above mentioned areas.

2.1 Tool - Life and its Characteristics

F. W. Taylor (43) was the first to investigate the tool - life relationships. The well known equation relating tool - life and cutting - speed does not, however, take into account the magnitude of the chip cross - sectional area and as such a different 'Taylor Constant' is required for each chip cross - sectional area. The necessary refinement in Taylor's equation has been done by Kronenberg (30). He derived initially an elementary and subsequently an extended cutting - speed law. The extended cutting - speed law considers the effect of both chip cross - sectional area and slenderness ratio, while the elementary law does not include the effect of the latter. Kronenberg, however, pointed out that chip

cross-sectional area has a considerable larger effect on the tool life than the slenderness ratio.

In a real-life situations tool-life rarely coincides with the predicted value. Yet, despite practical data on tool-life values, very little has been known about actual magnitude of the scatter. Various explainations have been forwarded to explain this scatter. Ermer and Wu (15, 16) have attributed the variance in tool-life purely to the experimental error. On the other hand, studies conducted by Wager and Barash (46) have led to the conclusion that the variation in tool-life can not be attributed to 'experimental error' but it is the resultant effect of many factors, material hardness, workpiece rigidity of mounting, thermo-electric effect, etc.. Although the effect of each individual factor is negligible, the sum total assumes appreciable proportion. The study also reveals that the variations in tool-life are inherent to the physical nature of the process which, like so many other physical processes, is stochastic. The authors analysed experimental data and found that the life of High Speed Steel (HSS) tools would be best represented by a positively skewed normal distribution.

Extensive efforts (17) have been directed to determine tool-life data for single-point tool but the available literature reveals that the tools of complex geometry (e.g., form tools) have received very little attention.

2.2 Economics of Machining

The work of Taylor (43) in the United States, culminating in a major paper in 1906, had an important influence on the development of economics of machining. He suggested that a relationship must exist between machining time and tool - life in order to have minimum cost of production. Till to - date this particular facet of metal removing has received concerted effort to establish conditions with a view to manufacture economically.

The discussion on the economics of machining is presented in a hierarchical order.

2.2.1 Single - Point Tool Economics

Many authors have performed experimental and analytical studies of the single point tool case to understand better how the cutting conditions affect production times and tool lives, and hence the costs of cutting. The problem consists in determining machining parameters in order to optimize a given objective function. The variety of objective functions in common vogue are :

1. minimum cost per piece,
2. minimum machining time per piece,
3. maximum profit per piece,
4. maximum profit rate, and
5. meeting of demand successfully, etc. .

The flood of literature in this field can be classified under two major sub-headings depending upon the type of formulation

and methodology used :

(a) Unconstrained Optimization, and

(b) Constrained Optimization.

(a) Unconstrained Optimization :

This optimization methodology has been adopted by a number of researchers including Taylor (43a), Brewer (7), Gilbert (22), Weill (47), Gilman (23) and Colding (8). They utilized empirical formulae to express tool - life as a function of various machining parameters such as cutting - speed, feed and depth of cut. The objective function is differentiated to obtain the required optima. These models are quite introductory in nature. However, these initial efforts triggered a lot of interest among the research workers to develop improved models.

Field, et al (19, 20) have taken tool - life data and have used a digital computer to generate tool - life equations. These equations were used to develop relationships for the determination of optimal speed. Only the first three criteria have been used in their analysis.

Wu & Ermer (48) concerned themselves with determining speed and feed so as to maximize profit. This was accomplished by assuming functions for marginal revenue and marginal cost. The optimum machining parameters were calculated by equating the

expressions. Armarego and Russel (1, 2) have argued that for a given income per component, profit made by the manufacturer per unit time is not optimum when the criterion of maximization of profit is used. Hence, from manufacturer's point of view without increasing the cost of commodity is of prime importance. A similar analysis has beem made by Okushima and Hitomi (35). Wu and Tee (50) have investigated the profit concept to obtain a better understanding of the profit region. The authors have found that optimum cutting speed seems to lie between minimum cost speed and the speed for meeting the demand satisfactorily.

Ermer and Faria - Gonzalez (14) have studied the sensitivity of machining cost with respect to the cutting conditions. This study was primarily conducted to understand the shape of the cost curve in the neighbourhood of the optimum cutting conditions.

In order to account for tool life scatter, Ermer and Wu (16) suggested that optimal cutting speed should not be defined uniquely and instead a range should be associated with it. They advocate the minimax principle for the purpose of speed and feed. Ermer and Morris (15) in another paper have recommended the use of a correction factor to account for uncertainties in tool - life. They claimed that the application of correction factor is more direct and simpler compared to the minimax principle.

Vul'fson and Deryabin (44) developed an iterative procedure for selecting the speed and feed which yields minimum

cutting cost. A computer based procedure was designed specifically for numerically controlled lathes.

Very few research workers have directed their efforts to machining operations other than simple turning. Armarego and Russel (2) has presented an analysis for single - pass shaping and milling processes based on maximum profit criterion.

b) Constrained Optimization

In a real life situation the choice of machining parameters is constrained by machine and cutting - tool characteristics and the component specifications. Hence, a constrained problem would represent machining activity in a more realistic way.

A linear programming formulation has been developed for the purpose of selection of a machine tool and corresponding cutting conditions such that a rotational part may be manufactured with the least cost in a job - shop environment (38). Krishna and Berra (28) has formulated a multi - pass face milling optimization model so as to minimize the cost of production. The model consists of a non - linear convex objective function constrained by linearized constraints. Rosen's Gradient Projection method is used to solve the problem.

Bhattacharya and Faria - Gonzalez (5) have presented a model considering surface finish as a principal constraining parameter. In their analysis the slack variables are introduced in the power form and the method of Lagrange Multiplier is used for ascertaining local minima.

In 1972 Iwata (25) proposed an analytical method applying a chance constrained programming concept to determine the optimum conditions introducing stochastic concept in both the objective function and the constraints. He has advocated that the constants and exponents in tool - life and cutting - force equation should be considered as stochastic parameters. It is shown that the optimum cutting conditions are significantly affected by these considerations.

2.2.2 Economics of Multi - Tool Set ups

This area remains, relatively, unexplored because of a few additional problems associated with it. One of the major problems in multi - tool set - ups is the identification of optimum groups of operations. The problem of grouping is irrelevant in the case of single-tool. The over all problem of multi - tool set up can also be divided into the following sub - problems :

a. Identification of optimum groups,

b. Determination of optimum cutting parameters, and

c. Determination of tool - replacement schedule.

The sub - problems are discussed in detail in the following sections.

a) Identification of Optimum Groups

A paper on the related subject of determining groups among the operations to be performed on a particular job is due to Jones and Morgan (27). They have employed a graphical technique

called Analysis of Interconnected Decision Areas originated by Luckman and Stringer (26). The method has no objective function and thus does not seek optimality, but rather it attempts to identify feasible alternatives when they exist.

Bartalucci (3) has formulated a dynamic programming model to solve the problem, but as number of operations increase the numerical effort required by the method increases enormously.

b) Determination of Optimum Cutting Parameters

Brown (7a) reviewed the literature and opined that when an operation comprises of two or more passes, all passes should be of same depth so as to achieve an optimum solution.

McCullough (32) has reported a very elementary and approximate analysis for multi - tool case. The equations derived were said to express the optimum tool - life for minimum cost and maximum production rate.

A detailed procedure to find optimum cutting conditions for multi - tool operation is presented by Goranskii (24) in his book 'Theory of Automation of Production Planning and of Tooling'. A major contribution of Goranskii lies in linearizing the model by taking logarithm of the both objective function and the constraints. To summarize, the algorithms put forward suffer from two major drawbacks. First, the tool - life equation has four constants and eight exponents which in real life situation are very

difficult to determine. The second reason for complexity of the algorithm is that the linear - program as initially developed, may be infeasible and the algorithms are required to adjust systematically to the various parameters until a feasible solution is obtained.

c) Determination of Tool Replacement Schedule

In machining operations involving many tools simultaneously, such as in multi - spindle machine tools, the choice of tool-change procedure and tool-change time interval is critical to the overall efficiency and economy of production. The study of the literature available reveals that the problem of determination of optimal cycle - length of a cutting-tool presumes machining parameters to be known.

A cost model for strict interval - replacement policy has been suggested by Taha (42). The objective function includes the cost of defective parts produced in addition to the tool - replacement costs. In order to simplify the model it has been assumed that any increase in percentage of defective items, over the situation when the process is stable, is caused only by a poor condition of the tools.

Ebert and Hershauer (12) has conducted a simulation experiment to examine cutting - tool operating costs under varying conditions. In their model the tool-replacement policies incorporate different type of quality control decision rules.

The strict interval - replacement policy has been found significantly better than the policies governed by quality control decision rules.

Okushima and Fujii (34) has formulated cost models for three different types of tool - replacement schedules.* The author has recommended that due to inherent uncertainties in the tool-life, the optimum solution can not be reached directly. Hence, a simulation method offers a useful means to determine an economical tool - change schedule. The computed data indicated that the scheduled tool - changes are more economical than the policy of replacing the tools as they fail.

Duncan (11) has developed a very sophisticated mathematical and numerical model for scheduling cutting tool - changes.

2.2.3 Group Technology and its Ramifications

Group technology is a technique of grouping similar components together in order to eliminate repeated efforts required to design and plan for a new component which resembles another component already designed.

Opitz (36) has developed a coding procedure for the jobs based upon the philosophy of **part** family manufacturing, production method that involves machining of parts in families.

---

* Explained in section 3.1.6 of Chapter III

Scott (41) has reported a method of automated planning for manufacturing which is called 'Regenerative Shop Planning'. This system put an end to the repetitive manufacturing decisions, redundant operation planning and the piling up of paper work. In this system the logic by which the original component was manufactured is placed in computer storage and utilized in planning of subsequent components. In order to institute this method the author has stated that one must be able to generate part families which have many characteristics in common.

As recently as August '73 Rajagopalan (39) has introduced a graph theory approach to the problem of grouping the machines in a batch - production shop into cells in such a way that each cell takes care of a family of similar components.

2.2.4 Automated Manufacturing Systems

Most of the work done in this area is of proprietary in nature, therefore, has not been reported in literature. Some of the well publicised systems are APT (4), MILMAP (4), AUTOPIT (4), EXAPT II (4), etc.. These systems allow the description of a part to be made in a language that closely resembles english. Berra and Barash, and Krishna (4, 29) have brought forward a comprehensive system for the automated planning and optimization of a manufacturing process. Their interest has centered on numerically controlled lathes, but the scope of their system ranges from the specification of the component to the numerical control

of manufacturing operation. Their work has laid foundation for automated planning but lacksin generality. The essential reason is that a numerically controlled lathe usually has a single tool which makes a series of cuts to generate a part, while multi - tool operation has got different characteristics of its own. Obviously single - tool aspects are embedded in multi - tool operation.

To summarize, the literature survey reveals the lack of scientific literature in the area of multi - tool operation. It also opens up a new awareness of the distance between the ultimate and the present.

CHAPTER III

GENERAL DESCRIPTION OF MULTI-TOOL OPERATION SYSTEMS

Before delving into the problem formulation proper, it is essential to have a better understanding of the system under study so as to help modelling. Hence, a description of some of the important elements synthesizing the system is delineated. The relevant portions of the model involving determination of cutting conditions, cutting time and production cost is presented in Chapter IV.

3.1 Elements

The major system elements of the production activity are :

1. The machine tool - a multi-spindle automatic,
2. The cutting tools,
3. The workpieces,
4. The machine operator,
5. The tool-setters and tool-setting activity,
6. The tool resharpening activity,
7. The tool failure, and
8. The tool changing activity.

Most of them are in common vogue and need no explanations. Discussion only whenever pertinent has been included.

3.1.1 The Machine Tool - a Multi-Spindle Automatic

The machining methods employed on multi - spindle automatics depend upon their principle of operation and on the type of blank used. Depending on the type of stock or blank, the machines can be classified as 'bar type' or 'chucking machines'. According to the principle of operation, the automatics are classified as parallel and progressive action machines. In the parallel action automatics the same operation is performed simultaneously on all the spindles. Thus, during one operating cycle as many workpieces are produced as there are spindles. In the progressive action automatic, each spindle with its blank or stock is indexed to the next position after machining is finished at each position. Every workpiece has to pass through each position consecutively.

In multi-spindle automatic the cycle time* is equal to the maximum of the minimum possible production time for each spindle. In such a situation the most expedient producing method is one in which the time required at each station is same.

Most models of multi - spindle automatics have common speed and feed for all the spindles. But the modern trend is to have different speeds and feeds for each spindle. In the present work this type of automatics are categorized as type - A. The previous type of automatics are categorized as type - B.

---

*Cycle time = machining time for the station requiring ma cutting time + tool approach time + indexin + tool withdrawal time + loading and unload time for the job.

3.1.2 The Machine Operator

The span of activity of the operator, generally, varies with shop conditions. It is not unusual to assign more than one machine to one operator in case of bar type automatics. The number of machines assigned to an operator is a function of : variety of activities that he has to perform, the expected frequency of machine requiring attention either due to tool failure or exhaustion of raw material, etc..

3.1.3 The Tool setters and Tool Setting Activity

This activity is critical in multi - tool operation. The practice of preset tooling is usually adopted to reduce down time. According to the predetermined replacement schedule the tool setter replenishes the machine with tools. During the setting time the production has to be stopped, so the downtime is strong function of the policy regarding the replacement schedule.

3.1.4 The Tool Resharpening Activity

The tools which are regrindable are sharpened to make them usable once again. The inventory of cutting-tool includes three echelons : the tool inventoried in the expense crib, (the new tool inventory and resharpened inventory), the tools inventoried at the machines, (machine float), and the tools stored in the grinding shop awaiting regrind, (regrind inventory).

Once the grind inspector receives a shipment of worn and damaged tools, he has to make several decisions. First, he decides which tools are to be reground and which are to be scrapped. Second, he may determine how much stock is to be ground from each tool. Finally, he determines whether or not the quality of reground tools is satisfactory. The grinders usually set their machines up so that one set-up will suffice for an entire lot. This means that the amount of stock removal for each lot depends upon the worst tool and so is the time of grinding. Many tools such as taps, small drills, throw away - tips, etc. are never reground.

3.1.5 The Tool Failure

The tool failure phenomena has been recognized as a probabilistic event. Inspite of the uncertainties involved, empirical relations have been proposed to find out expected values of tool life.

In this work, the production life of a cutting - tool is defined as the number of satisfactory workpieces produced from the time of tool - replacement to failure. It needs to be pointed out that whenever the tool is replaced before its failure, it is because of particular tool-replacement schedule. It is called premature removal. It will be discussed later in the section 3.1.6 that in some of the tool - replacement schedules premature removal is inevitable.

Duncan (11) has reported that the tool failure can be attributed to the aging effects; commonly categorised as wear, deterioration, fatigue, etc..

Lot of work is reported on the study of tool failure phenomena. Duncan has listed reasons for failure under two separate headings.

i) Sources dependent on tool, and

ii) Sources independent of tool.

Delving into all the pros and cons of the phenomena is not within the scope of the present research. It suffices to state that all the reasons grouped together make tool failure a random phenomena.

3.1.6 The Tool Changing Activity

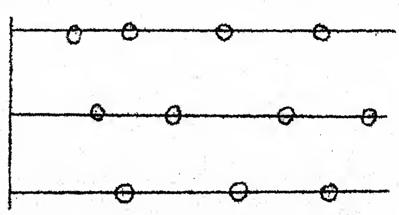
Tool changing is primarily a corrective action to maintain quality and avoid accidents due to loading of worn out tools. To take this action, the management must have proper guidelines for an economic tool - replacement schedule. As per Okushima and Fujii (34) tool-change procedures can be divided into the following categories:

1. Unscheduled tool - change : In this case, any individual tool will be changed whenever it fails. Schematically shown in Fig. 1.
2. Scheduled tool - change : In this case, there is a predetermined tool - change interval, but some unexpected tool failures may require extra tool changes. This alternative consists of two sub-alternatives.

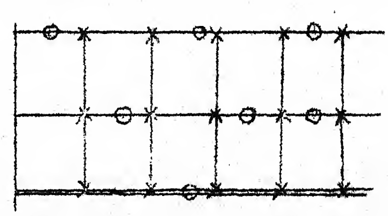
First, all tools will be changed without regard to the history of each tool during the preceding interval. This procedure is shown schematically in Fig. 2.

Second, only selected tools will be changed, that is, those tools which have been changed during the preceding interval will not be changed at this time. Only those tools which have not been changed will be replaced. This procedure is shown schematically in Fig. 3.

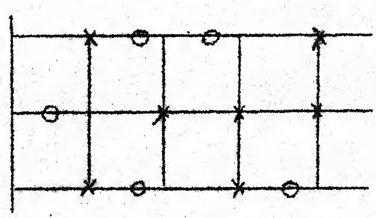
In the present work the two sub-alternatives are named as schedule 2-A and schedule 2-B respectively. In Fig. 1, Fig. 2, and Fig. 3, the example of machine tool with three spindles is given. The white circle in the figures represent the changes occuring due to tool failure and the crosses represent the tool-change at the predetermined tool-change time. Here, it can be observed that in case of scheduled tool-change procedure, the premature tool removal may occur quite often.



Unscheduled Tool-Replacement
FIG. 1



Scheduled Tool-Replacement 2-A
FIG. 2



Scheduled Tool-Replacement 2-B
FIG. 3

Any tool changing activity is associated with downtime production losses. Often, the costs associated with these downtimes are intangible in nature. The time required for tool changing normally depends only upon the relative position of the tool on slides, tool inventories and quality control tests after the setting of the tool. The change - over time can be split into :

Time for,

(1) machine shut down,

(2) tool retrieval,

(3) tool removal and replacement, and

(4) quality control testing.

CHAPTER IV

PROBLEM FORMULATION

4.1 The Problem

It is required to have a method of determining an optimal strategy for the machining operations performed on multi-spindle automatics. This involves the machine tool selection, tool-replacement schedule and the determination of machining parameters - speed and feed rate for every tool. The objective is to minimize either the total cost of production for a given lot size or total machining time.

The problem requires the following types of data as input :

1. Job specification as specified by the customer,
2. Operation plan of the job to be manufactured. The operation plan specify - (a) groups among the operations, and (b) assignment of spindles to the various groups formed.
3. The relevant data generated from job drawing and its specifications,
4. The data about cutting - tools used include,
   - (a) the type of the tool used,
   - (b) the mean and standard deviation of the tool - life and force equations exponents for each tool under consideration, and
   - (c) the relevant characteristics of the tools used.

5. Machine type and its characteristics,

6. Cost and time parameters, and

7. Size of the lot to be produced and the attrition allowance.

Chapter VI contains a detailed listing of various characteristics for all types of data mentioned above.

The following sections provide a comprehensive description of the various models formulated to solve the problem. The formulation is based on the assumption that the optimization is carried out only for primary operations like, turning, drilling, form turning, parting off, etc. The operations like, thread cutting, chamfering, knurling, reaming, etc. are taken as secondary operation. These operations, if at all to be performed on a job, are excluded from the optimization routine.

Separate models are formulated and studied for two types of automatics - (i) multi - spindle automatic for which different speeds and feeds are available for each spindle termed as Type-A, and (ii) the machine tool having common speed and feed value for all the spindles termed as Type-B.

4.2 The Objective Function

The objective functions for optimization of machining operation in common vogue are :

1. Minimization of cost per unit,
2. Minimization of production time per unit,
3. Maximization of profit per unit,

4. Maximization of profit rate, and

5. Meeting the demand successfully.

The profit function can not be justified because of its subjective nature. A distinct estimation of contribution of each machining operation to the total profit earned is very difficult. Hence, the decisions based on the profit function may involve high degree of arbitraryness and the whole optimization efforts loose their value.

The demand meeting function may be appropriate only for a particular shop - condition : a shop may have to pay a very high penalty for making delays in the shipment.

Therefore, the present work restricts itself to study the following two objective functions. The objective functions are :

1. Minimization of total machining time. In the multi - spindle automatic operation the cycle time dictates the total machining time. Therefore, minimization of machining time is equivalent to minimization of cycle time, and

2. Minimization of cost of production for a given lot size.

4.2.1 Cycle Time Minimization

The various components constituting cycle time can be listed as follows :

1. maximum of machining times for all the cutting tools, minutes,

2. t_w - time required for withdrawal of all the tools, minutes,

3. t_I - time required in indexing of the spindles, minutes,

4. t_l - time required for unloading and loading, minutes, and

5. t_a - time required for approach of the tools, minutes.

The cycle time, TC, in minutes can be expressed as

$$TC = t_l + t_a + \max_{i,j} (t_{ij}) + t_w + t_I \qquad (4.1)$$

$$i = 1, 2, \ldots, N$$

$$j = 1, 2, \ldots, n_i$$

where

i - index for spindle,

j - operation index for operation on i^{th} spindle,

t_{ij} - machining time of an operation, minutes,

N - number of spindles, and

n_i - total number of operations performed on i^{th} spindle.

As it is pointed out above that t_l, t_a, t_w, and t_I time elements are constant depending upon a combination of machine tool and the job to be produced, hence need not be considered in the objective function. Thus the objective function reduces to,

$$\text{minimize } TC = \max_{i,j} (t_{ij}), \quad i = 1, 2, \ldots, N, \quad j = 1, 2, \ldots, n_i \qquad (4.2)$$

In the following sections the cycle time will be understood equal to its variable part only.

The machining time of an operation can be expressed in terms of machining parameters as follows :

$$t_{ij} = \frac{l_{ij}}{V_i \; f_{ij}} \tag{4.3}$$

where

l_{ij} - tool travel length for j^{th} operation on i^{th} spindle, inches

V_i - i^{th} spindle speed, rpm., and

f_{ij} - feed rate on i^{th} spindle for j^{th} operation, inch/rev.

It is to be noticed that the subscripts for speed and feed rate are meaningful only in the case of automatic type - A, but the subscripts do not affect the formulation for the automatic of type - B. The affect of machine type will be realized only in the development of constraint set.

4.2.2 Production Cost Minimization :

The major components contributing to the total cost of machining operation are : machine and operating cost, tool costs and non productive costs. The machining cost is that of operator and overhead computed over the production time for a lot to be produced. The tool cost is comprised of tool edge cost, machine

setting cost, intermediate tool - changing cost, and tool regrinding cost. Finally, the non productive cost includes, the production lost in non-cutting time due to intermediate settings because of tool failure and the cost of operator for the same period.

In the following section the expression for various costs are developed for the two tool replacement procedures; 1) unscheduled tool - replacement (Fig. 1), Case 1, 2) scheduled tool-replacement (2-A) (Fig.2), Case 2. In Case - 1 there will be unexpected tool changes and may become expensive and time consuming in comparison to scheduled ones. But excessively frequent scheduled changes are also uneconomical.

Case 1 - Unscheduled Tool-Replacement :

Machining and Operating Cost

The total machining and operating cost, C_m, in rupees can be expressed as :

$$C_m = TC \cdot NJ \cdot (C_1 + C_2) \tag{4.4}$$

where

- NJ - total number of jobs to be produced (the lot includes the attrition allowance),
- C_1 - direct labour cost, Rs./min., and
- C_2 - overhead rate, Rs./min..

Tool Cost

The uncertainties involved in tool - life prohibits one to know in advance the moments of tool failures. In such a case

expected values can be worked out only. The expected value of tool - life depends upon machining parameters. Hence, the selection of machining parameters should prolong the tool - failure and reduce tool replacement costs.

Vyoskovoskii (45) has reported that the case of unscheduled replacement is same as series replacement. A system comprised of many components of average finite life has an average life whose reciprocal is equal to the sum of reciprocals of life of individual components. In the case of multi - spindle automatic the machine tool can be thought of as a system and the components of the system are various cutting - toolshaving a finite mean life. In the present work,the machine tool failure is ignored. Hence, the average life, $\bar{\bar{T}}$, of the system can be expressed as

$$\frac{1}{\bar{\bar{T}}} = \sum_{i=1}^{N} \sum_{j=1}^{n_i} \frac{1}{T_{ij}} \tag{4.5}$$

where

T_{ij} - mean life of j^{th} cutting tool on the i^{th} spindle.

A cost expression for the system is next formulated. The formulation has been carried out with the assumption that the various downtimes enumerated in the section 3.1.6 of Chapter III are combined into one single term - tool changing time.

The expression for the total tool cost, CCH, is written as

$$CCH = \sum_{i=1}^{N} \sum_{j=1}^{n_i} \left[\frac{p_{ij}}{m_{ij}+1} + 1 \right] \cdot EC_{ij} + \sum_{i=1}^{N} \sum_{j=1}^{n_i} \left(\left[p_{ij} + 1 \right] - \left[\frac{p_{ij}}{m_{ij}+1} + 1 \right] \right) \cdot TG_{ij} \cdot CG + \left[p_{ij} \right] \cdot TS_{ij} \cdot CS + TS \cdot C_3 \qquad (4.6)$$

where

$$p_{ij} = \frac{TP_{ij}}{T_{ij}}$$

CG - grinding labour rate, Rs./min.,

TS - initial machine setting time, min.,

C_3 - machine tool setter rate, Rs./min., and

$[Z]$ - indicates that greatest integer value of Z is to be taken.

For the i^{th} spindle:

TP_{ij} - total actual machining time for j^{th} tool $= \dfrac{L_{ij} \cdot NJ}{V_i \cdot f_{ij}}$, min.,

L_{ij} - actual machining length with tool j, inches,

m_{ij} - number of possible regrinds with tool j,

EC_{ij} - cost of a new j^{th} tool - edge, Rs.,

TG_{ij} - grinding time for a cutting tool j, minutes, and

TS_{ij} - intermediate tool - change time for j^{th} tool, min.,

Non-Productive Cost

The non-productive loss occurs whenever the machine is stopped either for the purpose of tool - change or because of machine

maintenance. In literature it has been found that this cost is accounted in terms of direct labour wage rate. However, the loss incurred is not only the money paid to the labour but there is a production loss also during the machine stoppages. The loss of production can be accounted by imagining of the possible output during the idle time. Therefore, the loss of production due to the down time can be considered as lost profit. Since, this factor is highly nebulous and the formulation assumes that this information will be provided by the management. Hence, the non - productive cost, CNP, is expressed as follows :

$$CNP = \sum_{i=1}^{N} \sum_{j=1}^{n_i} \left(\frac{TS_{ij}}{TC} \right) \cdot \left[p_{ij} \right] \cdot CP + \sum_{i=1}^{N} \sum_{j=1}^{n_i} \left[p_{ij} \right] \cdot TS_{ij} \cdot C_1 \qquad (4.7)$$

where

CP - expected profit per unit, Rs.

The total cost of production, CT, is summation of equations (4.4), (4.6), and (4.7). Therefore,

$$CT = C_m + CCH + CNP \qquad (4.8)$$

<u>Case 2 - Scheduled Tool - Replacement</u>

Scheduled tool - replacement is characterized by pre-determined time interval after which all the tools are replaced irrespective of their condition. Moreover, if any tool fails before the predetermined tool - replacement interval, it is also replaced

at the time of its failure. The knowledge of probability distribution function (p.d.f.) of tool-life can help in the determination of total number of replacements before the scheduled time of tool-change. But very little experimental work has been reported in literature to establish the exact nature of p.d.f. of the tool-life. However, the p.d.f. will be a composite distribution obtained by taking weighted mean of the distribution of different modes of tool failure; weightages being assigned on the basis of relative dominance of each mode of failure. The unavailability of tool-life distribution for various types of failures compels one to adopt a simple but approximate method for the purpose. In this study a factor C_{ij} for j^{th} tool on i^{th} spindle is defined by the following equation :

$$C_{ij} = 1 - \frac{T_{ij} - q\,\sigma_{T_{ij}}}{SI} \tag{4.9}$$

where

$-2 \leqslant q \leqslant 2$

SI - tool-replacement interval, number of components produced during the interval and

$\sigma_{T_{ij}}$ - standard deviation of tool-life of j^{th} tool on i^{th} spindle.

There are three possible values of C_{ij}'s. They are :

1. $0 \leqslant C_{ij} \leqslant 1$,
2. $C_{ij} > 1$, and
3. $C_{ij} < 0$.

The first possibility does not need any modification. The later two possibilities are modified to $C_{ij} = 1$ and $C_{ij} = 0$ respectively. $C_{ij} < 0$ implies that there are very remote chances of tool-failure and $C_{ij} > 1$ implies the certainty of tool failure before the scheduled hour.

Thus, the number of failures of a tool, NR_{ij}, is determined by the multiplication of C_{ij} and NS; the number of scheduled intervals. NS and NR_{ij} can be expressed mathematically as :

$$NS = \left[\frac{TC \cdot NJ}{SI}\right] \tag{4.10}$$

$$NR_{ij} = \left[NS \cdot C_{ij}\right] \tag{4.11}$$

Thus, the cost expressions are developed on the basis of the above assumption.

Machine and Operating Cost

This cost component is written in the same way as in Case 1. Therefore,

$$C_m = TC \cdot NJ \cdot (C_1 + C_2) \tag{4.12}$$

Tool Cost

In case of scheduled tool-replacement cost is incurred due to two types of tool-change activity :

a) tool changing cost because of tool-failure before the scheduled interval, CCH_1, and

b) the cost because of scheduled group replacement, CCH_2.

The expressions for CCH_1 and CCH_2 are written as follows :

$$CCH_1 = \sum_{i=1}^{N} \sum_{j=1}^{N_i} \left(\left[NR_{ij} + 1 \right] - \left[\frac{NR_{ij}}{m_{ij} + 1} + 1 \right] \right) . TG_{ij} . CG + \sum_{i=1}^{N} \sum_{j=1}^{n_i} \left[\frac{NR_{ij}}{m_{ij} + 1} + 1 \right] . EC_{ij} + \sum_{i=1}^{N} \sum_{j=1}^{n_i} \left[NR_{ij} \right] . TS_{ij} . CS + TS . C_3 \tag{4.13}$$

$$CCH_2 = \sum_{i=1}^{N} \sum_{j=1}^{N_i} \left[\frac{NS}{m_{ij} + 1} + 1 \right] . EC_{ij} + \sum_{i=1}^{N} \sum_{j=1}^{n_i} \left(\left[NS \right] - \left[\frac{NS}{m_{ij} + 1} + 1 \right] \right) . TG_{ij} . CG + \sum_{i=1}^{N} \sum_{j=1}^{n_i} \left[NS \right] . TS_{ij} . CS \tag{4.14}$$

The tool cost, CCH, is sum of CCH_1 and CCH_2.

Non-Productive Cost

This cost component occurs for both (a) and (b) type of tool-change activity involved and the expression is written keeping in mind the same assumptions as in Case 1. Therefore,

$$CNP = \sum_{i=1}^{N} \sum_{j=1}^{n_i} \left[\frac{TS_{ij}}{TC} \right] . CP . NR_{ij} + \left[\frac{GRT}{TC} \right] . NS . CP + \sum_{i=1}^{N} \sum_{j=1}^{n_i} TS_{ij} . NR_{ij} . C_1 + GRT . NS . C_1 \tag{4.15}$$

where

GRT - time taken for replacing all tools together, min.

Total cost can be expressed as the summation of equations (4.12), (4.13), (4.14) and (4.15).

4.3 Constraints

The total cost of machining, thus determined, should be optimized with respect to machining parameters i.e., speed and feed rate. However, the choice of these parameters is restricted by the number of constraints which result from consideration of the physical process of machining and limitation of the machine tool under consideration. For example, the spindle speed and feed must lie within the range of maximum and minimum speeds and feeds available on the machine tool; the cutting force should not exceed the maximum force that can be resisted by the work and tool holding devices; the cutting power required should be less than the net power available from the driving motor of the machine tool.

The machining parameters determine the tool-life, the cutting-forces and the power consumed. In most of the literature on metal-cutting, the coefficients and the exponents involved in tool-life equation and force equation are considered to be deterministic. Iwata (25) has advocated that the deterministic values for the constants and the exponents are insufficient for the purpose of estimating tool-life and cutting forces experienced by the

tool. In the present work the stochastic characteristic of the exponents and the constants are taken into account with the help of chance constrained programming concept as suggested by Bracken and McCormick (6). The concept unriddles the problem of converting probabilistic constraints into deterministic ones. In brief, the concept is as follows :

"An optimization problem can be stated as :

Determine x_j's $(j = 1, 2, \ldots, n)$ to minimize the objective function

$$F = F(x_1, x_2, \ldots, x_n)$$

subject to the constraints

$$\sum_{j=1}^{n} a_{ij} \; x_j \geq b_i \, , \qquad i = 1, 2, \ldots, m$$

where

a_{ij}'s are the technological coefficients and,

b_i's are the constraint coefficients.

In this formulation a_{ij}'s and b_i's are random variables with normal distribution, while F is deterministic function of x_j's. Such a problem can be solved by the chance constrained programming concept.

Under the chance constraints

$$\text{Prob}\left(\sum_{j=1}^{n} a_{ij} \; x_j - b_i \right) \geq p_i \, , \quad i = 1, 2, \ldots, m \tag{4.16}$$

find x_j's to minimize the objective function

$$F = F(x_1, x_2, \ldots, x_n)$$

where p_i's in equation (4.16) are predetermined probability levels at which the corresponding constraints must be satisfied.

The chance constraint in equation (4.16) is equivalent to the following deterministic constraint as derived in appendix A :

$$\sum_{j=1}^{n} \bar{a}_{ij} x_j + \psi(p_i) \Big(\sum_{j,k=1}^{n} \rho_{a_{ij} a_{ik}} \sigma_{a_{ij}} \sigma_{a_{ik}} x_j x_k$$

$$- 2 \cdot \sum_{j,k=1}^{n} \rho_{a_{ij} b_i} \sigma_{a_{ij}} \sigma_{b_i} x_j + \sigma_{b_i}^2 \Big)^{1/2} \geqslant \bar{b}_i \; , \; i = 1, 2, \ldots, m \qquad (4.17)$$

where

$$p_i = \frac{1}{2\pi} \int_{\psi(p_i)}^{\infty} e^{-t^2/2} \, dt \; ,$$

$\bar{a}_{ij}$ and $\bar{b}_i$ are the mean values of a_{ij} and b_i respectively,

$\sigma_{a_{ij}}$ and σ_{b_i} are the standard deviation of a_{ij} and b_i respectively, and

$\rho_{a_{ij} a_{ik}}$ and $\rho_{a_{ij} b_i}$ are correlation coefficients between a_{ij} and a_{ik} and between a_{ij} and b_i respectively.

The chance constrained problems is, thus, reduced to the equivalent non-linear programming problem to minimize the objective function f under the deterministic non-linear constraints in equation (4.17)."

The constraining factors considered are summarized as follows :

(1) Constraints determined by machine-tool specifications;

(a) the maximum cutting speed constraint,

(b) the minimum cutting speed constraint,

(c) the maximum feed constraint, and

(d) the minimum feed constraint.

(2) Constraints determined by machine tool dynamics;

(e) the allowable maximum cutting-force constraint; and

(f) the allowable maximum power consumption.

(3) Constraints determined by tool-life consideration;

(g) tool life should be greater than or equal to time required to produce a number of jobs. The number is also to be determined.

(4) Constraints determined by the cycle time;

(h) the machining time of a cutting-tool shoud be less than the cycle time.

Cutting-Force and Power

The allowable cutting-force may concern the tool-work deflection, temperature generated at tool-work interface and so on. However, in the present work, the values of maximum allowable force are chosen arbitrarily. The cutting-force and power consumed are expressed in terms of cutting parameters as follows :

$$FC = 10^{\alpha_0} S^{\alpha_1} f^{\alpha_2} d^{\alpha_3} \quad (4.18)$$

$$PC = \frac{FC \cdot S}{33000} \quad (4.19)$$

where

S - cutting speed, fpm,

f - feed rate, feed/revolution,

d - depth of cut,

FC - cutting-force, lbs,

PC - power, HP,

α_0, α_1, α_2, and α_3 are normally distributed parameters with $\bar{\alpha}_0$, $\bar{\alpha}_1$, $\bar{\alpha}_2$, and $\bar{\alpha}_3$ as their mean value and σ_{α_0}, σ_{α_1}, σ_{α_2} and σ_{α_3} as their standard deviations respectively.

Hence, the chance constraints for cutting-force and power become

$$\text{prob}\ (FC \leqslant FC_{max}) \geqslant p_1 \quad \text{and} ,$$

$$\text{prob}\ (PC \leqslant PC_{max}) \geqslant p_2 \quad \text{respectively.}$$

where

FC_{max} - Maximum force allowable,

PC_{max} - Maximum power available, and

p_1, p_2 are predetermined confidence levels.

Tool Life Constraints

The empirical relation for tool - life is taken of the form

$$T_L = 10^{\alpha_4} S^{\alpha_5} f^{\alpha_6} d^{\alpha_7} \quad (4.21)$$

where

T_L - tool life, min.

α_4, α_5, α_6, and α_7 are considered as normally distributed parameters with $\bar{\alpha}_4$, $\bar{\alpha}_5$, $\bar{\alpha}_6$, and $\bar{\alpha}_7$ as their mean values and σ_{α_4}, σ_{α_5}, σ_{α_6}, and σ_{α_7} as their standard deviation respectively.

The tool-life constraint is imposed because it is highly desirable that a freshly reground tool last as long as possible. However, since tool failure phenomena is inherently probable, the constraint can never be satisfied with absolute certainty. Hence, the chance constraint can be expressed as

$$\text{prob}\ (T_L \geqslant NP \cdot \text{machining time}) \geqslant p_3$$

where

p_3 - predetermined confidence level, and

NP - number of components to be machined before the tool failure.

Cycle Time Constraints

The constraints regarding cycle time are presented in the form as suggested by Fox (22). The constraints take shape as follows :

$$t_{ij} - TC + \epsilon \geqslant 0 , \quad i = 1, 2, \ldots N, \quad j = 1, 2, \ldots n_i, \tag{4.22}$$

ϵ is a small positive number and it is essential to introduce it to avoid any constraint becoming equal to zero.

If ϵ is omitted, then at least one constraint will be active and the suggested solution algorithm will stop working.

The constraints (a, b, c, d, and h) are deterministic while the constraints (e, f, and g) are chance constrained and they can be converted into deterministic with the help of equation (4.17). For simplicity the correlation coefficients in equation (4.17) are assumed to be zero. The original constraints after logarithmic transformation of variables are linear, however, the converted constraints are non-linear because of the nature of conversion applied. The resulting constraints are expressed as follows :

Maximum speed constraint -

$$g_{1i} = \ln (V_{max}) - x_{1i} \geqslant 0 \tag{4.23}$$

Minimum speed constraint -

$$g_{2i} = -\ln (V_{min}) + x_{1i} \geqslant 0 \tag{4.24}$$

Maximum feed constraint -

$$g_{3ij} = \ln (f_{max_{ij}}) - x_{2ij} \geqslant 0 \tag{4.25}$$

Minimum feed constraint -

$$g_{4ij} = -\ln (f_{min_{ij}}) + x_{2ij} \geqslant 0 \tag{4.26}$$

Maximum force constraint -

$$g_{5ij} = \ln (FC_{max_{ij}}) - \bar{\alpha}_{oij} (\ln 10.) - \bar{\alpha}_{1ij}\, x_{1i} - \bar{\alpha}_{2ij}\, x_{2ij} - \bar{\alpha}_{3ij} \ln (d_{ij}) + \psi(p_{1ij})\, (g'_{5ij})^{1/2} \geqslant 0 \tag{4.27}$$

$$g'_{5ij} = \sigma^2_{\alpha_{oij}} (\ln 10.)^2 + \sigma^2_{\alpha_{1ij}} x_{1i}^2 + \sigma^2_{\alpha_{2ij}} x_{2ij}^2$$

$$+ \sigma^2_{\alpha_{3ij}} (\ln d_{ij})^2 \qquad (4.28)$$

Tool-Life Constraint -

$$g_{6ij} = -\ln (L_{ij}) + \bar{\alpha}_{4ij} (\ln 10.) + (1 + \bar{\alpha}_{5ij})\, x_{1i}$$

$$+ (1 + \alpha_{6ij})\, x_{2ij} + \alpha_{7ij} (\ln d_{ij}) - \ln (NP)$$

$$+ \psi (p_{3ij}) (g'_{6ij})^{1/2} \geqslant 0 \qquad (4.29)$$

$$g'_{6ij} = \sigma^2_{\alpha_{4ij}} (\ln 10.)^2 + \sigma^2_{\alpha_{5ij}} x^2_{1i} + \sigma^2_{\alpha_{6ij}} x^2_{2ij}$$

$$+ \sigma^2_{\alpha_{7ij}} (\ln d_{ij})^2 \qquad (4.30)$$

Cycle Time Constraint -

$$g_{7ij} = -\ln (L_{ij}) + x_{1i} + x_{2ij} + \ln (TC) + \epsilon \geqslant 0 \qquad (4.31)$$

Number of Components Constraint -

$$g_8 = -\ln (NP) + \ln (NJ) \geqslant 0 \qquad (4.32)$$

where

$$x_{1i} = \ln (V_i) , \quad x_{2ij} = \ln (f_{ij})$$

The above constraints are for all $i = 1, 2, \ldots , N$ and $j = 1, 2, \ldots , n_i$.

In the above constraint set the index i refers to spindle and the index j is for an operation on i^{th} spindle. It is to be

noticed that the above constraint set is written in a general form applicable to both type of automatics under consideration.

The constraint set presented above does not include the power constraint. It is because the structure of this constraint will be different for two types of automatics under consideration. The horse power constraint for the automatic type - A is as follows:

$$g_{9i} = \ln (PC_{max_i}) - \sum_{j=1}^{n_i} \ln PC_{ij} + \psi (p_{2i}) (g'_{9i})^{1/2} \geqslant 0 \tag{4.32}$$

$$g'_{9i} = \sum_{j=1}^{n_i} (PC_{ij})^2 . \left(\ln (10.)^2 . \sigma^2_{\alpha 1ij} + x^2_{1i} . \sigma^2_{\alpha 2ij} + x^2_{2ij} . \sigma^2_{\alpha 3ij} + (\ln d_{ij})^2 . \sigma^2_{\alpha 4ij} \right) \tag{4.33}$$

for all $i = 1, 2, \ldots, N$.

Hence, there will be as many constraints of type (4.32) as number of spindles.

The horse power constraint for the automatic type - B :
In this case there will be only one constraint.

$$g_9 = \ln .(PC_{max}) - \sum_{i=1}^{N} \sum_{j=1}^{n_i} \ln(PC_{ij}) \div \psi (p_2) (g'_9)^{1/2} \geqslant 0 \tag{4.34}$$

$$g'_9 = \sum_{i=1}^{N} \sum_{j=1}^{n_i} (PC_{ij})^2 . \left((\ln 10.)^2 . \sigma^2_{\alpha oij} + x^2_{1i} . \sigma^2_{\alpha 1ij} + x^2_{2ij} . \sigma^2_{\alpha 2ij} + (\ln d_{ij})^2 \sigma^2_{\alpha 3ij} \right) \tag{4.35}$$

The total number of constraints, for example, for a case of five spindles automatic type - A and n_i vector (2, 1, 1, 1, 1,), will be 44, while, for type - B the number is 30. The variables correspond to cutting parameters. This number will be 11 and 4 in two types of automatics respectively for the above example.

Thus, one finds that the problem reduces to an optimization problem with non - linear objective function constrained by non-linear constraints. The available mathematical programming techniques ensure of a global minima only for the convex functions. But both the objective functions under considerations are not convex and none of them is having a expression for partial derivatives, therefore, any optimum solution obtained can not be claimed as global minima.

Chapter V discusses the solution algorithm of penalty function approach.

CHAPTER V

OPTIMIZATION PROCEDURE

5.1 Introduction

The optimization problem formulated in Chapter IV in its generalized form can be stated as :

$$\text{Minimize} \quad f(x)$$

$$\text{s.t.} \quad g_i(x) \geqslant 0 , \quad i = 1, 2, \ldots, m$$

where x is the vector of independent variables $(x_1, x_2, \ldots, x_n)$ corresponding to cutting parameters. The constraints form a bounded region in n - dimensions.

A number of non-linear programming methods such as the gradient projection method (40), Zoutendizk's feasible direction method (51), penalty function method (18), etc. have been discussed in the literature. Most of the suggested methods are of fairly recent origin (1958 and onwards) and a fool proof answer to the problem of selection of a method is not yet established. However, the experiences of different. research workers is always of help. Fox (22) reports that the gradient projection method has been described for the general non-linear programming problem, but its effectiveness is mainly limited to problems in which the constraints $g_i(x)$ are linear-functions of x . Lasdon (31) opines that interior penalty function approach is particularly attractive in dealing with problems that have markedly non-linear constraints and the objective function

since, it approaches the solution value from inside the constraint set, and thus, avoids the difficulty of movement along non-linear boundaries of the set, which is required in the other methods. Hence, the interior penalty function method is adopted in this study.

5.2 Optimization Methodology

The optimization procedure consists of converting the general constrained problem to a sequence of unconstrained problems. The sequence of such problems are solved by Davidon - Fletcher - Powell's Variable Metric Method (21). Golden Search is employed to carry out one variable minimization.

5.2.1 Conversion to Unconstrained Minimization

Fiacco and McCormick (18) have developed an algorithm for transforming the general constrained problem involving inequality constraints to a sequence of unconstrained minimization problems. The transformation defines a new function $\phi(\bar{x}, r)$,

$$\phi(\bar{x}, r) = f(\bar{x}) + r \cdot \sum_{i=1}^{m} \frac{1}{g_i(\bar{x})} \tag{5.1}$$

and the minimization is carried over a strictly monotonic decreasing sequence of r - values.

Fiacco and McCormic have shown that if

(i) the interior of the constraint set is non empty,

(ii) the function $f(\bar{x})$ and $g_i(\bar{x})$, $(i = 1, 2, \ldots, m)$ should have second derivatives continuous,

(iii) the set of points in the constraint set for which $f(x) \leqslant V_o$ is bounded for every finite V_o, and

(iv) the function $f(\bar{x})$ is bounded below for x in the constraint set,

then, the optimal solution to the unconstrained problem approaches a local minima of the constrained problem as the value of r approaches zero. If, in addition

(v) $f(\bar{x})$ and $g_i(\bar{x})$, $(i = 1, 2, \ldots, m)$ are convex functions, and

(vi) $\phi(\bar{x}, r)$ is strictly convex in the interior of the constraint set for every $r > 0$,

then, the optimal solution to the unconstrained problem approaches the absolute minimum of the constrained problem as r approaches zero.

The steps comprize of :

1. Start with $\bar{x}_o$, an initial solution vector, which must be strictly inside the constraint set, and $r_i > 0$. Let $i = 1, 2, \ldots, n$.
2. Minimize $\phi(\bar{x}_i, r_i)$ starting from $\bar{x}_{i-1}$, and subject to no constraints.
3. If the convergence criteria is satisfied, go to Step 4, otherwise reduce r by choosing $r_{i+1} < r_i$ and return to Step 2 with i replaced by i + 1.

4. Take $r = 0$ and minimize the objective function and the iteration is terminated when the convergence criteria are satisfied in the unconstrained minimization routine.

There are number of points to be considered in applying the method :

a) The starting solution $\bar{x}_o$ required by Step 1 has to be carefully chosen, i.e., not very far from optima to avoid excessive computational efforts.

b) Care has to be exercised in the initial value for r.

c) The convergence criteria in Step 3 has to be carefully selected.

In the present problem, x_o has been chosen by a careful judgement. The convergence criteria chosen is simple. The Step 3 is terminated when the difference in two successive functional values is relatively small. This is a weak criterion - a better one being a check on the difference between the values of $\bar{x}_{i+1}$ and $\bar{x}_i$. However, in the present work the first criteria is considered.

For the selection of initial value of r, a method suggested by Fiacco and McCormick is explained in Appendix B.

5.2.2 Procedure for Unconstrained Minimization

An unconstrained minimization of the function $\phi(x, r)$ must be carried out over the design vector $\bar{x}$ for each value of r used in Fiacco - McCormic algorithm. The method makes use of the

quadratically convergent iterative descent method structured by Davidon and subsequently modified by Fletcher and Powell (21).

The method uses the concept of local Hessian J_i^{-1}, but approximates it by a metric H_i. The procedure for computing the metric completely eliminates the need for evaluating second derivatives and performing matrix inversions, and yet the metric which is improved at each iteration converges to J_m^{-1}.

The steps in the algorithm are :

1. Start with a positive definite matrix H_o (usually chosen as the identity matrix) and an initial point $\bar{x}_o$. The i^{th} step, i = 0, 1, 2 ... proceeds as follows :
2. Compute the gradient vector $\nabla \phi(\bar{x}_i, r) = G_i$
3. Compute the direction $\bar{S}_i = -H_i \nabla \phi(\bar{x}_i, r)$
4. Choose a step length α^*_i to minimize

   $$F(\alpha) = \phi(x_i + \alpha_i S_i, r)$$
5. Compute $\bar{\sigma}_i = \alpha^*_i \bar{S}_i$
6. Compute a new value $\bar{x}_{i+1}$ from the relationship

   $$\bar{x}_{i+1} = \bar{x}_i + \alpha^*_i \bar{S}_i$$
7. Compute $\nabla \phi(\bar{x}_{i+1}, r) = G_{i+1}$
8. Compute $\bar{y}_i = G_{i+1} - G_i$
9. Compute the matrix A_i

   $$A_i = \bar{\sigma}_i \bar{\sigma}_i^T / \bar{\sigma}_i^T \bar{y}_i$$

10. Compute the matrix B_i

$$B_i = -H_i \bar{y}_i \bar{y}_i^T H_i / \bar{y}_i^T H_i \bar{y}_i$$

11. Compute the successor

$$H_{i+1} = H_i + A_i + B_i$$

12. Check for the convergence criteria, if it is satisfied, return to constrained minimization routine. Otherwise, return to Step 2 using the successor matrix as the new H_i and replace i by i + 1.

In the above algorithm the gradient vector (calculated in Steps 2 and 7) is found out. The method of finite differences is employed here and is discussed in the Appendix C. The Step 4 is explained in section 4.2.3 of this chapter.

In applying the method care has been exercised to ensure that the H - matrix should always be positive definite and it is not updated with data arising from poor approximation to α_i^*. Fox has mentioned many approaches to this problem. First, the algorithm used for computing α_i^* may be reapplied until $S_i^T G_{i+1}$ is sufficiently small, another alternative is to simply skip the updating Steps 9, 10 and 11 of the algorithm when $S^T G_{i+1}$ is too large. In other words if α_i^* is not close enough to minimum along S_i, $H_{i+1} = H_i$ and $S_{i+1} = H_{i+1} G_{i+1}$ and the process is continued.

It is difficult to choose between these approaches. The first method may require excessive computations to refine α_i^*

at points far from x_i, but the second approach may skip valuable opportunities to improve the H matrix. A compromise is struck at this state; α_i^* is calculated twice and if value of $S_i^T\ G_{i+1}$ is still large, the updating steps are skipped. In the present work another method suggested by Fox is employed. A number $C = S_i^T G_{i+1} / S_i^T G_i$ is calculated. The contention is that $C = 0$ at points of minimum. At all points along S_i in the neighbourhood of α_i^*, $C \ll 1$. The value of C chosen is .001. So if the calculated $C > .001$, the updating steps are skipped.

Another area of numerical difficulty is the classical round off problem. If the scaling is bad; depending on the scale factor, $H_1 = H_o + B_o$ or $H_1 = A_o$ and the latter form is singular. The remedies are : increase precision of arithmatic or a proper scaling of the variables, or continuous check on H matrix. The last method is put to use in the programme. The remedy consists of setting H matrix to H_o or some other predetermined positive definite matrix and the process is continued. Thus a check is continuously kept on the factor $G_i^T\ H_i\ G_i$. Any time it is positive, the matrix H is reassigned the value of identity matrix. If $G_i^T\ H_i\ G_i$ is at positive level, it implies that the direction $-H_i\ G_i$ is no longer a direction of descent and H matrix has to be modified.

In the algorithm S_i are normalised to avoid excessive increment in design variables at any iteration. It is possible because in the present problem, the variables are transformed by

taking logarithm while $\alpha_i S_i$ is at normal scale. The normalisation consists of $S_i = S_i / (S.S)^{1/2}$.

The termination criterias consists of check on step size α_i and $\alpha_i S_i$. If any of these have a magnitude less than a pre-assigned ϵ, ($\epsilon > 0$, $\epsilon << 1$), the process is terminated.

5.2.3 Golden Search Algorithm (18)

The numerical difficulty experienced with inaccurate α_i^* has already been discussed. The field of uni-directional minimization with all the sophistication attained remains a comparatively crude area. Golden Search method which is used in the present work is one of the best methods available but still relies on arbitrary but so called Fibonacci number (18).

The steps are as follows :

1. First an upper bound θ_u is obtained. The first lower bound $\theta_L = 0$. θ_u is obtained by evaluating the function $\phi(x, r)$ at successive points whose θ values are in the Fibonacci ratio, $1.618 = (1 + \sqrt{5}) / 2$; that is $\theta_u = \sum_{j=0}^{\bar{J}} (1.618)^j$

   where $\bar{J}$ is the smallest non negative integer j such that $\phi\left[x_i + \sum_{k=0}^{j} (1.618)^k S_i\right] \geq \phi(x_i)$.

2. The interval boundary $\bar{\theta}$ is reduced by computing two values of θ in the interval $(\theta_u - \theta_L)$,

$$\theta_a = \theta_L \div .382 \; (\theta_u - \theta_L)$$

$$\theta_b = \theta_L + .618 \; (\theta_u - \theta_L)$$

3. The ϕ values of the (interior) correspond to θ_a abd θ_b are compared.

4. If $\phi(x_i + \theta_a S_i, r) < \phi(x_i + \theta_b S_i, r)$, then $\theta_L \leqslant \bar{\theta} \leqslant \theta_b$, because of the property that .382/ 0.618 = 0.618, by letting $\theta_u' = \theta_b$, $\theta_b' = \theta_a$ and $\theta_L' = \theta_L$ and recomputing $\theta_a' = \theta_L' + .382\;(\theta_u' - \theta_L')$. Step 3 can now be repeated.

5. If $\phi(x_i + \theta_a S_i, r) > \phi(x_i + \theta_b S_i, r)$, then by letting $\theta_L' = \theta_a$, $\theta_a' = \theta_b$, and $\theta_u' = \theta_u$ and computing $\theta_b' = \theta_L' + .618\;(\theta_u' - \theta_L')$, Step 3 can be repeated.

6. If $\phi(x_i + \theta_a S_i, r) = \phi(x_i + \theta_b S_i, r)$, Let $\theta_L' = \theta_a$, $\theta_u' = \theta_b$ and repeat Step 2.

7. When $\theta_u - \theta_L$ is acceptably small, $\bar{\theta}$ is approximated by $\bar{\theta} = \theta_L + \theta_u / 2$ and $x_{i+1} = x_i + \bar{\theta} S_i$.

The procedure for obtaining the optimum value as outlined above ensures a global optimum only for convex functions. For functions not satisfying the criterion the minimum obtained may be but a local minimum only (31).

CHAPTER VI

COMPUTERIZATION AND TESTING OF THE MODEL

6.1 Computer Package

The optimization procedure explained in Chapter V has been computerized so that it may be applied to the practical situation of machining of parts on multi - spindle automatics. The program also generates the constraint set (equations 4.23 to 4.35). The generation of constraint subroutine is quite general. The objective functions, 1) minimization of cycle time and 2) minimization of total cost of production, have been programmed. The computer listing is put in appendix D. The comment cards are put liberally to explain the working of the program and the meaning of various important variables used in the program.

In the case of multi-spindle automatics the amount of information to be fed to the Computer is enormous compared to the single-tool case. To a great extent this difficulty is overcome by representing the information in a matrix form. This method is advantageous also in many ways : 1) development of a generalized computer package is possible, 2) use of the package becomes easy, 3) the development of logic is simplified, etc..

In this work, an attempt has been made to develop a general purpose package. The optimum machining parameters can be determined for any operation in which metal is removed in form of chips. For example, single tool operation on lathe,

turret lathe operation, milling operation, etc. can be handled. However, the package has been tested only for the multi-spindle automatic operation. The package can handle an automatic having upto 8 spindles. A maximum of 6 operations can be performed on each spindle. The machine tool specification determines the size of the job which the machine can handle.

The package has been tested for a sample job shown in Fig. 4. The input information needed for the determination of optimum cutting parameters is given below.

6.2 Operation Plan of the Job shown in Fig. 4.

Spindle 1 - Stepped turning to produce surface (2) and (3).

Spindle 2 - Form turning to produce (4) and (5) surfaces.

Spindle 3 - Form turning to produce surface (6) and spot drilling.

Spindle 4 - Drilling of hole 1/4".

Spindle 5 - Parting off and thread cutting.

6.3 The Input

The information regarding a metal removing activity can be grouped under the following headings :

1. Raw material characteristics,
2. Component requirements,

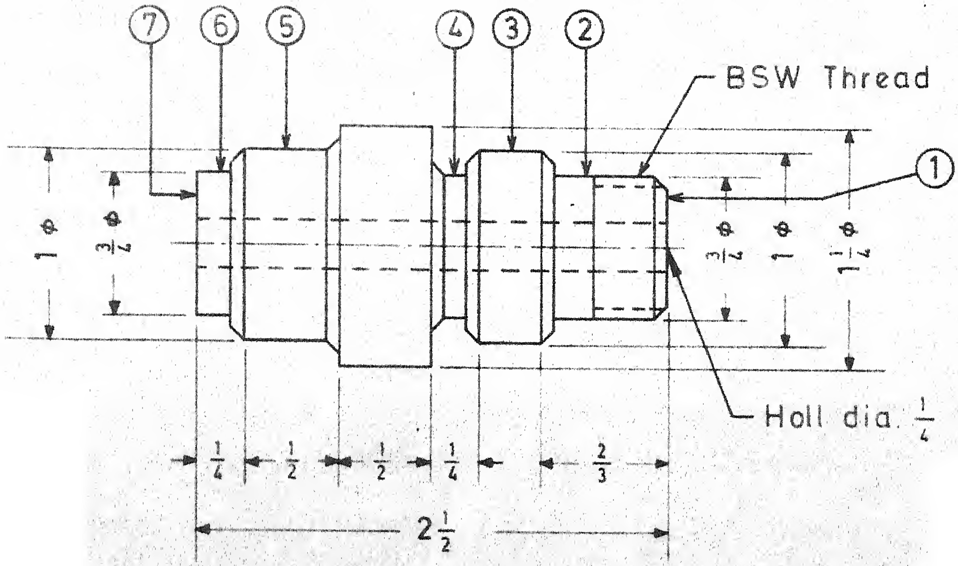


Fig. 4.

All dimensions are in Inches.

All chamfers $1/_{16} \times 45^{\circ}$

Circled numbers show surface number.

3. Data about the component to be machined,
4. Machine characteristics,
5. Tool characteristics,
6. Mean and standard deviation of tool-life and cutting-force equation exponents, and
7. Cost and time parameters.

In the following sections the various elements pertaining to a data group along with their representative values taken in the present work are described.

6.3.1 Raw Material Characteristics

This group of information is named as Raw Material Characteristics Matrix denoted by AMATR. Table 1 shows the FORTRAN name and the element along with its value.

TABLE 1

Raw Material Characteristic Matrix

FORTRAN NAME	Description	Value
AMATR (1)	Type of material used (Code No.)	1060
AMATR (2)	Brinell hardness of the raw stock	250
AMATR (3)	Yield strength of the raw material, psi	127500
AMATR (4)	Any important dimension (length, diameter, etc.) of the material, inches.	1.2
AMATR (5)	Raw material cost rate, Rs. per/unit	1.00

6.3.2 Component Requirements

The component requirements are stored in Component Requirement Matrix denoted as COMR. The various elements are put in Table 2 along with their assumed values.

TABLE 2

Component Requirement Matrix

FORTRAN NAME	Description	Value
COMR (1)	Number of Components to be manufactured.	10000
COMR (2)	Material grade specification (Code No.)	1060
COMR (3)	Raw Material length required for component, inch.	2.6
COMR (4)	Attrition percentage.	5.0
COMR (5)	Date of Receiving the order.	
COMR (6)	Due date	

6.3.3 Data about the Component

This information is put under the matrix, Component Data Matrix, designated by COMD. This is a three dimensional matrix. The first dimension is allocated to the spindle, the second is for operations, and the third is for various elements.

The specific values have been generated from the job (Fig. 4) under consideration. The description of the input elements is as follows :

The subscript I refers to spindle and J to an operation on I^{th} spindle.

1. COMD (I, J, 1) - Tool travel length, inches.
2. COMD (I, J, 2) - Depth of cut, inches. This is specified only in the case of multi - spindle operation where in the job is finished in one cycle of the machine. Otherwise, it is a machining parameter to be determined, inches.
3. COMD (I, J, 3) - Feed index. It is a logical variable which governs the construction of feed constraint. For example, if the two tools are employed on the same slide, the feed constraints are to be formed only for any one tool. Its value is 1 if new feed constraints are to be formed. Otherwise it is 2.
4. COMD (I, J, 4) - Speed index. It is also a logical variable governing the speed constraints. For example, if the two tools are put on the same spindle, the speed in rpm for both the operations has to be the same. The constraint is generated only for one tool. It is 1 if new constraint set is to be generated. Otherwise it is 2.
5. COMD (I, J, 5) - Machining length, inches. This value indicates the actual machined length by a tool. In some

cases it may be different than tool travel length (COMD (I, J, 1)), inches.

6. COMD (I, J, 6) - Index for secondary operation or primary operations. This is a binary variable. It is 0 for a secondary operation and 1 for a primary operation.

7. COMD (I, J, 7) - Depth of cut index. This index helps in deciding about whether the depth of cut is a parameter to be determined or it is a given parameter. It's value is 0 and 1 for the two cases respectively.

8. COMD (I, J, 8) - It is an index to indicate whether the operation is to be done on the axial slide or cross slide of the automatic. This binary number also helps in constructing the constraint set. For example, if on a particular station the two operations are being done, one operation on axial slide and the other on cross slide, the feed rate can be different for axial slide than for the cross slide. It is 0 if the operation is done on axial slide, otherwise it is 1.

TABLE 3

COMPONENT DATA MATRIX

		(COMD (I, J, K), K = 1, 8)							
Spindle No.	Opn. No.	1 inch	2 inch	3	4	5 inch.	6	7	8
I = 1	J = 1	1.250	0.125	1	1	1.250	1	1	1
	J = 2	1.250	0.125	2	2	.670	1	1	1
I = 2	J = 1	0.500	0.870	1	1	.500	1	1	1
I = 3	J = 1	-	-	-	-	-	0	-	-
	J = 2	0.125	0.250	1	1	.125	1	1	1
I = 4	J = 1	2.500	.125	1	1	2.500	1	1	0
I = 5	J = 1	0.125	.440	1	1	.125	1	1	1
	J = 2	-	-	-	-	-	0	-	-

6.3.4 Machine Characteristics

This group of information is fed to the computer under the matrix name Machine Characteristic Matrix (MCM). It is a two dimensional array. The first subscript relates to the number of machine under consideration and the second subscript is for various elements. The table 4 shows the description of various elements. Here, the characteristics of one machine are shown, because only one machine is taken and the sensitivity analysis is carried out for its horse power capacity only.

TABLE 4

MACHINE CHARACTERISTIC MATRIX

FORTRAN NAME I=1	Description	Value
CMC (I, 1)	Number of stations on the machine	5
CMC (I, 2)	Number of axial slides	5
CMC (I, 3)	Type of machine (Type - A or Type - B)	(1, 0)
CMC (I, 4)	Maximum speed available, rpm	2400
CMC (I, 5)	Minimum speed available, rpm	100
CMC (I, 6)	Minimum feed rate, ipr.	.0005
CMC (I, 7)	Maximum feed rate, ipr.	.2000
CMC (I, 8)	Minimum depth of cut, inch	.0100
CMC (I, 9)	Maximum depth of cut, inch	.3000
CMC (I, 10)	Horse power	20.0000
CMC (I, 11)	Power Transmission efficiency	.8000
CMC (I, 12)	Axial minimum feed rate, inch/rev.	.0005
CMC (I, 13)	Axial maximum feed rate, inch/rev.	.2000

6.3.5 Tool Characteristics

A cutting tool, generally, associates itself with a large number of specifications. In the present work only the relevant characteristics are specified. This group of information is put under Tool Characteristic Matrix denoted as TCM. The description of elements is given as follows.

1. TCM (I, J, 1) - Number of teeth on the tool.
2. TCM (I, J, 2) - Diameter of the tool (if relevant), inch.
3. TCM (I, J, 3) - Maximum force or torque allowable lbs. or lbs. in. respectively.
4. TCM (I, J, 4) - Number of regrinds permissible
5. TCM (I, J, 5) - Tool edge cost, Rs., and
6. TCM (I, J, 6) - Horse power constant.

The values corresponding to the elements of matrix TCM is shown in Table 5.

TABLE 5

Tool Characteristics Matrix

		TCM (I, J, K), K = 1, 6					
Spindle No.	Opn. No.	1	2 in.	3 lbs.	4	5 Rs.	6 X 10^5
I = 1	J = 1	1	-	200	3	2	.795
	J = 2	1	-	200	3	2	.795
I = 2	J = 1	1	-	200	3	10	.795
I = 3	J = 1	-	-	-	-	-	-
	J = 2	-	-	100	3	5	.795
I = 4	J = 1	2	.25	50	3	3	7.950
I = 5	J = 1	1	-	50	3	5	.795
	J = 2	-	-	-	-	-	-

6.3.6 Exponents of Tool-Life and Cutting-Force Equations

The mean values of the exponents of tool-life and cutting-force equation for the cutting tools involved in the example are taken from Kronenburg (30). The standard deviations of the exponents are chosen arbitrarily.

The mean values and standard deviation of exponents for tool-life and cutting force equation are put in Table 7 and 6 respectively.

TABLE 6

Cutting-Force Equation Exponents

Spindle No.	Operation No.	MEAN VALUES				STANDARD DEVIATION				Material
		$\bar{\alpha}_0$	$\bar{\alpha}_1$	$\bar{\alpha}_2$	$\bar{\alpha}_3$	σ_{α_0}	σ_{α_1}	σ_{α_2}	σ_{α_3}	
I = 1	J = 1	3.9	0.0	0.8	1.2	0.4	0.0	0.2	0.2	HSS
	J = 2	3.9	0.0	0.8	1.2	0.4	0.0	0.2	0.2	HSS
I = 2	J = 1	3.9	0.0	0.8	0.0	0.4	0.0	0.2	0.2	HSS
I = 3	J = 1	-	-	-	-	-	-	-	-	-
	J = 2	3.9	0.0	0.8	0.0	0.4	0.0	0.2	0.2	HSS
I = 4	J = 1	3.9	0.0	0.9	0.0	0.8	0.0	0.2	0.2	HSS
I = 5	J = 1	3.9	0.0	0.8	0.0	0.4	0.0	0.2	0.2	HSS
	J = 2	-	-	-	-	-	-	-	-	-

6.3.7 Time and Cost Parameters

The representative values of cost parameters are as follows :

C_1 - Direct labour cost rate = .02 Rs./min.

C_2 - Overhead cost rate = .01 Rs./min.

C_3 - Machine setting cost rate = .05 Rs./min.

CG - Grinding labour cost rate = .03 Rs./min.

CS - Tool-change cost rate = .05 Rs./min.

Time parameters are different for different tools. The elements are grouped under matrix TIMEC. The description of the elements is as follows :

TIMEC (I, J, 1) - Tool - Change time, min., and

TIMEC (I, J, 2) - Tool - grinding time, min.

TABLE 7

Tool-Life Equation Exponents

Spindle No.	Operation No.	MEAN VALUES				STANDARD DEVIATION			
		$\bar{\alpha}_4$	$\bar{\alpha}_5$	$\bar{\alpha}_6$	$\bar{\alpha}_7$	$\sigma_{\alpha 4}$	$\sigma_{\alpha 5}$	$\sigma_{\alpha 6}$	$\sigma_{\alpha 7}$
I = 1	J = 1	8.9	-4.2	-1.8	-0.5	0.25	0.20	0.05	0.03
	J = 2	8.9	-4.2	-1.8	-0.5	0.25	0.20	0.05	0.03
I = 2	J = 1	9.5	-4.2	-1.8	-0.5	0.25	0.20	0.05	0.03
I = 3	J = 1	-	-	-	-	-	-	-	-
	J = 2	9.9	-4.2	-1.8	-0.5	0.25	0.20	0.05	0.03
I = 4	J = 1	10.3	-4.2	-1.8	-0.5	0.25	0.20	0.05	0.03
I = 5	J = 1	9.9	-4.2	-1.8	-0.5	0.25	0.20	0.05	0.03
	J = 2	-	-	-	-	-	-	-	-

The table 8 shows the values of the elements in matrix TIMEC.

TABLE 8

Time Parameters

Spindle No.	Operation No.	TIMEC (I, J, 1) Min.	TIMEC (I, J, 2) Min.
I = 1	J = 1	20.0	5.0
	J = 2	20.0	5.0
I = 2	J = 1	25.0	10.0
I = 3	J = 1	-	-
	J = 2	25.0	10.0
I = 4	J = 1	15.0	20.0
I = 5	J = 1	10.0	10.0
	J = 2	-	-

The above set of data is fed to the computer as part of the main program. The program generates the constraint set and evaluates the objective function. The initialization of all the cutting parameters is done in the constraint generation program. Subsequently the logic enters into the SUMT algorithm and finds out the optimum solution.

CHAPTER VII

RESULTS AND CONCLUSIONS

7.1 The Results and Conclusions

The results obtained for the example (Fig. 4) described in Chapter VI are discussed here.

The optimum cutting conditions and the corresponding values for the objective functions are obtained. The optimization is carried out with respect to the minimization of cycle time and minimization of total production cost. The results are tabulated in Tables 9 - 12. The tables 9 and 10 correspond to the minimization of cycle time for a machine tool having capacity of 20 and 10 horse power respectively. The tables 11 and 12 correspond to the production cost minimization and show the machining parameters and total cost incurred for unscheduled and scheduled tool-replacement (2-A) procedures respectively.

Both type of automatics - the automatic having different speed and feed on each spindle (Type-A) and the automatic having common speed and feed for each spindle (Tybe-B) have been studied. The results for both of them are tabulated in Tables 9 - 12.

The nomenclature used for the rows of tables 9 - 12 is explained in Table 8a.

The efforts are directed to study the effect of different probability levels concerning the constraints (equations

TABLE 8a

Table Of Nomenclature For Tables 9, 10, 11, and 12

x11 - Feed rate for first spindle, ipr.

x12 - Feed rate for second spindle, ipr.

x13 - Feed rate for third spindle, ipr.

x14 - Feed rate for fourth spindle, ipr.

x15 - Feed rate for fifth spindle, ipr.

x21 - Speed for first spindle, rpm.

x22 - Speed for second spindle, rpm.

x23 - Speed for third spindle, rpm.

x24 - Speed for fourth spindle, rpm.

x25 - Speed for fifth spindle, rpm.

NP - Number of components for which the tool-life constraint is to be satisfied.

t11 - Travel-time for tool 1 on spindle 1, minutes.

t12 - Travel time for tool 2 on spindle 1, minutes.

t21 - Travel time for tool 1 on spindle 2, minutes.

t31 - Travel time for tool 1 on spindle 3, minutes.

t41 - Travel time for tool 1 on spindle 4, minutes.

t51 - Travel time for tool 1 on spindle 5, minutes.

CT - Cycle time, minutes

T11 - Life of tool 1 on spindle 1, minutes.

T12 - Life of tool 2 on spindle 1, minutes.

T21 - Life of tool 1 on spindle 2, minutes.

Table 8a (Continued)

T31	-	Life of tool 1 on spindle 3, minutes.
T41	-	Life of tool 1 on spindle 4, minutes.
T51	-	Life of tool 1 on spindle 5, minutes.
NS	-	Number of tool-replacement schedules in scheduled tool-replacement procedure.
SL	-	Life of the system, unscheduled tool-replacement procedure, minutes.
PT	-	Total machining time, minutes.
PC	-	Cost corresponding to machining time, Rs.
TCC	-	Tool cost in case of unscheduled tool-replacement procedure, Rs.
TCC1	-	Tool cost in case of scheduled tool-replacement for the tools changed because of tool failure before the scheduled interval.
TCC2	-	Tool change cost due to group replacement in scheduled tool-replacement procedure, Rs.
ITC	-	Idle time cost corresponding to TCC; Rs.
ITC1	-	Idle time cost corresponding to TCC1, Rs.
ITC2	-	Idle time cost corresponding to TCC2, Rs.
TC	-	Total cost of production, Rs.

TABLE 9

Cycle Time Minimization -
Horse Power 20

	AUTOMATIC TYPE-A HORSE POWER=20				AUTOMATIC TYPE-B HORSE POWER=20.0			
	$p_1 = p_2 = p_3 = .5000$	$p_1 = p_2 = p_3 = .8413$	$p_1 = p_2 = p_3 = .9772$	$p_1 = p_2 = p_3 = .9987$	$p_1 = p_2 = p_3 = .5000$	$p_1 = p_2 = p_3 = .8413$	$p_1 = p_2 = p_3 = .9772$	$p_1 = p_2 = p_3 = .9987$
1. $x11 \times 10^3$	7.9	19.7	14.1	19.1	5.6	7.3	11.2	12.1
2. $x12 \times 10^3$	4.3	3.4	2.7	4.3	5.6	7.3	11.2	12.1
3. $x13 \times 10^3$	1.3	1.2	17.9	3.4	5.6	7.3	11.2	12.1
4. $x14 \times 10^3$	7.8	15.8	8.3	20.2	15.2	16.2	21.7	22.1
5. $x15 \times 10^3$	2.1	2.1	1.4	3.9	5.6	7.3	11.2	12.1
6. $x21 \times 10^{-1}$	78.0	56.9	81.7	145.7	108.9	145.3	239.9	239.9
7. $x22 \times 10^{-1}$	61.8	85.0	112.4	167.7	108.9	145.3	239.9	239.9
8. $x23 \times 10^{-1}$	92.9	142.9	105.1	134.5	108.9	145.3	239.9	239.9
9. $x24 \times 10^{-1}$	126.9	119.1	178.8	239.9	108.9	145.3	239.9	239.9
10. $x25 \times 10^{-1}$	79.1	124.2	156.0	161.6	108.9	145.3	239.9	239.9
11. $NP \times 10^{-1}$	20.5	37.4	47.0	25.1	20.0	20.0	20.0	20.0
12. $t11 \times 10^1$	2.0	1.1	1.1	0.4	2.0	1.2	0.5	0.5
13. $t12 \times 10^1$	2.0	1.1	1.1	0.4	2.0	1.2	0.5	0.5
14. $t21 \times 10^1$	1.3	1.2	1.2	0.5	0.6	0.3	0.1	0.1
15. $t31 \times 10^1$	1.7	1.1	0.1	0.4	0.3	0.2	0.1	0.1
16. $t41 \times 10^1$	2.5	1.3	1.7	0.5	1.5	1.1	0.5	0.5
17. $t51 \times 10^1$	1.8	1.2	1.4	0.5	0.5	0.3	0.1	0.1
18. $CT \times 10^1$	2.5	1.3	1.7	0.5	2.0	1.2	0.5	0.5
19. $T11 \times 10^1$	62.8	84.8	35.2	4.6	22.9	7.6	1.2	1.1
20. $T12 \times 10^{-1}$	62.8	84.8	35.2	4.6	22.9	7.6	1.2	1.1
21. $T21 \times 10^{-1}$	93.3	41.2	20.3	4.1	37.0	12.4	1.9	1.8
22. $T31 \times 10^{-1}$	143.1	28.3	9.6	15.5	108.1	36.1	5.5	5.4
23. $T41 \times 10^{-1}$	51.2	36.7	16.7	3.4	20.0	7.7	1.3	1.2
24. $T51 \times 10^{-1}$	187.0	46.9	30.8	12.9	111.3	37.1	5.6	5.5

TABLE 10

Cycle Time Minimization -
Horse Power 10

	AUTOMATIC TYPE-A HORSE POWER=10.0				AUTOMATIC TYPE-B HORSE POWER=10.0			
	$p_1 = p_2 = p_3$ =.5000	$p_1 = p_2 = p_3$ =.8413	$p_1 = p_2 = p_3$ =.9772	$p_1 = p_2 = p_3$ =.9987	$p_1 = p_2 = p_3$ =.5000	$p_1 = p_2 = p_3$ =.8413	$p_1 = p_2 = p_3$ =.9772	$p_1 = p_2 = p_3$ =.9987
1. $x11 x 10^{3}$	7.9	19.7	14.1	19.1	5.6	7.3	11.2	12.1
2. $x12 x 10^{3}$	4.2	3.4	2.7	4.3	5.6	7.3	11.2	12.1
3. $x13 x 10^{3}$	1.3	1.2	1.8	3.4	5.6	7.3	11.2	12.1
4. $x14 x 10^{3}$	7.8	15.8	8.3	20.2	15.2	16.2	21.7	22.1
5. $x15 x 10^{3}$	2.1	2.1	1.4	3.9	5.6	7.3	11.2	12.1
6. $x21 x 10^{-1}$	78.4	56.9	81.7	145.7	108.9	145.3	239.9	239.9
7. $x22 x 10^{-1}$	61.0	85.0	112.4	167.7	108.9	145.3	239.9	239.9
8. $x23 x 10^{-1}$	83.9	142.9	105.1	134.5	108.9	145.3	239.9	239.9
9. $x24 x 10^{-1}$	128.2	119.1	178.8	239.9	108.9	145.3	239.9	239.9
10. $x25 x 10^{-1}$	80.3	124.2	156.1	161.6	108.9	145.3	239.9	239.9
11. $NP x 10^{-1}$	19.8	37.4	47.1	25.1	20.0	20.0	20.0	20.0
12. $t11 x 10^{1}$	2.0	1.1	1.1	0.4	2.0	1.2	0.5	0.5
13. $t12 x 10^{1}$	2.0	1.1	1.1	0.4	2.0	1.2	0.5	0.5
14. $t21 x 10^{1}$	1.4	1.2	1.2	0.5	0.6	0.3	0.1	0.1
15. $t31 x 10^{1}$	1.7	1.1	0.1	0.4	0.3	0.2	0.1	0.1
16. $t41 x 10^{1}$	2.5	1.3	1.7	0.5	1.5	1.1	0.5	0.5
17. $t51 x 10^{1}$	1.8	1.2	1.4	0.5	0.5	0.3	0.1	0.1
18. [illegible]$x 10^{1}$	2.5	1.3	1.7	0.5	2.0	1.2	0.5	0.5
19. $T11 x 10^{-1}$	62.0	84.8	35.3	4.6	22.9	7.6	1.2	1.1
20. $T12 x 10^{-1}$	62.0	84.8	35.3	4.6	22.9	7.6	1.2	1.1
21. $T21 x 10^{-1}$	99.1	41.2	20.3	4.1	37.0	12.0	1.9	1.6
22. $T31 x 10^{-1}$	138.4	28.3	9.6	15.5	108.1	36.1	5.5	5.4
23. $T41 x 10^{-1}$	49.5	36.7	16.8	3.4	20.0	7.7	1.3	1.2
24. $T51 x 10^{-1}$	179.8	46.9	30.8	12.8	111.3	37.2	5.6	5.5

TABLE 11

Total Production Cost Minimization - Unscheduled Tool-Replacement Policy

	AUTOMATIC TYPE-A, HORSE POWER=10.0 NUMBER OF COMPONENTS TO BE PRODUCED = 10000				AUTOMATIC TYPE-B, HORSE POWER=10.0 NUMBER OF COMPONENTS TO BE PRODUCED = 20000			
	$p_1 = p_2 = p_3 = .5000$	$p_1 = p_2 = p_3 = .8413$	$p_1 = p_2 = p_3 = .9772$	$p_1 = p_2 = p_3 = .9987$	$p_1 = p_2 = p_3 = .5000$	$p_1 = p_2 = p_3 = .8413$	$p_1 = p_2 = p_3 = .9772$	$p_1 = p_2 = p_3 = .9987$
1. $x11 \times 10^3$	5.8	5.6	7.9	8.3	12.8	50.3	131.8	142.1
2. $x12 \times 10^3$	2.2	2.1	2.4	2.7	12.8	50.3	131.8	142.1
3. $x13 \times 10^3$	1.8	1.5	2.4	2.5	12.8	50.3	131.8	142.1
4. $x14 \times 10^3$	7.5	6.7	6.0	6.9	23.7	121.9	178.7	181.0
5. $x15 \times 10^3$	2.2	1.4	2.4	2.8	12.8	50.3	131.8	142.1
6. $x21 \times 10^{-1}$	75.3	78.1	51.5	48.9	34.1	27.9	25.2	24.1
7. $x22 \times 10^{-1}$	56.5	48.8	44.1	43.0	34.1	27.9	25.2	24.1
8. $x23 \times 10^{-1}$	61.3	54.4	57.1	55.2	34.1	27.9	25.2	24.1
9. $x24 \times 10^{-1}$	104.7	121.9	134.6	137.5	34.1	27.9	25.2	24.1
10. $x25 \times 10^{-1}$	87.5	92.1	201.9	210.0	34.1	27.9	25.2	24.1
11. $NP \times 10^{-1}$	38.3	66.7	57.5	61.7	28.4	24.0	20.6	20.2
12. $t11 \times 10^{-1}$	2.9	2.8	3.2	3.1	2.9	0.9	0.4	0.4
13. $t12 \times 10^{1}$	2.9	2.8	3.2	3.1	2.9	0.9	0.4	0.4
14. $t21 \times 10^{1}$	2.8	3.4	3.3	3.1	0.8	0.2	0.1	0.1
15. $t31 \times 10^{1}$	1.7	2.4	1.7	1.6	0.4	0.1	0.1	0.1
16. $t41 \times 10^{1}$	3.2	3.0	3.1	3.1	3.1	0.7	0.6	0.6
17. $t51 \times 10^{1}$	1.5	2.4	0.6	0.5	0.7	0.2	0.1	0.1
18. $OT \times 10^{1}$	3.2	3.4	3.3	3.1	3.1	0.9	0.6	0.6
19. $T11 \times 10^{-1}$	88.6	80.5	251.3	-	464.8	312.3	209.6	199.3
20. $T12 \times 10^{-1}$	88.6	80.5	251.3	-	464.8	312.3	209.6	199.3
21. $T21 \times 10^{-1}$	203.1	330.1	408.2	-	751.3	504.7	338.9	321.9
22. $T31 \times 10^{-1}$	294.1	496.2	289.6	-	992.9	998.1	989.9	971.5
23. $T41 \times 10^{-1}$	96.1	64.6	524.7	-	540.2	296.9	307.7	295.3
24. $T51 \times 10^{-1}$	133.0	161.4	9.2	-	987.9	927.6	981.7	985.3
25. $SL \times 10^{-1}$	22.0	20.9	-	-	142.1	90.1	67.8	65.0
26. $PT \times 10^{-2}$	35.2	37.2	36.3	36.1	68.2	19.8	13.2	13.2
27. PC	105.6	112.2	108.9	108.3	204.6	59.4	39.6	39.6
28. TCC	30.6	36.2	36.2	36.2	28.7	27.0	27.0	27.0
29. ITC	89.9	92.5	57.4	57.4	16.8	0.0	0.0	0.0
30. TC	261.5	278.3	239.8	238.0	318.3	106.2	79.2	79.8

TABLE 12

Total Production Cost Minimization - Scheduled Tool-Raplacement Policy (2-A)

	AUTOMATIC TYPE-A HORSE POWER=10.0 -2 q 2				AUTOMATIC TYPE-B HORSE POWER=10.0 -2 q 2			
	$p_1 = p_2 = p_3$ .5000	$p_1 = p_2 = p_3$ .8413	$p_1 = p_2 = p_3$ .9772	$p_1 = p_2 = p_3$ .9987	$p_1 = p_2 = p_3$ .5000	$p_1 = p_2 = p_3$ .8413	$p_1 = p_2 = p_3$ .9772	$p_1 = p_2 = p_3$ .9987
1. x11x10^{3}	5.4	4.2	3.8	3.1	4.2	4.5	6.0	6.5
2. x12x10^{3}	1.8	1.3	1.1	1.9	4.2	4.5	6.0	6.5
3. x13x10^{3}	2.2	1.5	1.1	1.6	4.2	4.5	6.0	6.5
4. x14x10^{3}	5.0	4.4	4.4	7.4	6.0	6.7	7.3	8.2
5. x15x10^{3}	1.8	1.3	1.9	1.3	4.2	4.5	6.0	6.5
6. x21x10^{-1}	77.0	66.0	71.2	57.0	40.3	52.6	61.1	92.2
7. x22x10^{-1}	60.3	64.8	84.3	54.0	40.3	52.6	61.1	92.2
8. x23x10^{-1}	62.4	88.1	110.9	73.2	40.3	52.6	61.1	92.2
9. x24x10^{-1}	67.6	74.6	84.7	59.0	40.3	52.6	61.1	92.2
10. x25x10^{-1}	62.4	88.4	55.4	77.0	40.3	52.6	61.1	92.2
11. NPx10^{-1}	63.7	333.3	648.1	720.1	635.6	710.5	806.7	905.2
12. t11x10^{1}	3.0	4.5	4.5	4.3	7.4	5.3	3.4	2.5
13. t12x10^{1}	3.0	4.5	4.5	4.3	7.4	5.3	3.4	2.5
14. t21x10^{1}	3.1	4.0	3.8	3.4	2.1	1.5	1.0	0.8
15. t31x10^{1}	1.4	1.4	1.6	1.6	1.1	0.8	0.5	0.4
16. t41x10^{1}	7.4	7.6	6.7	5.7	10.3	7.1	5.6	4.5
17. t51x10^{1}	2.6	2.6	2.9	3.0	1.8	1.3	0.8	0.7
18. CTx10^{1}	7.4	7.6	6.7	5.7	10.3	7.1	5.6	4.5
19. T11x10^{-1}	88.6	169.7	146.3	235.7	635.9	264.1	133.3	-
20. T12x10^{-1}	86.6	169.7	146.3	235.7	635.9	264.1	133.3	-
21. T21x10^{-1}	189.0	192.9	101.6	263.6	981.2	426.8	215.4	-
22. T31x10^{-1}	236.6	107.0	68.1	179.5	983.3	961.8	629.4	-
23. T41x10^{-1}	513.6	412.5	278.3	596.2	896.9	359.6	211.6	-
24. T51x10^{-1}	442.3	193.7	618.3	326.1	997.3	918.1	647.7	-
25. NS	34	6	3	3	3	3	2	2
26. PTx10^{-2}	162.8	167.2	148.4	125.4	226.6	156.2	123.2	99.0
27. PC	488.4	501.6	445.2	376.2	679.8	468.6	369.6	297.0
28. TCC1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
29. ITC1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
30. TCC2	539.3	108.0	54.0	54.0	54.0	54.0	45.5	45.5
31. ITC2x10^{-1}	118.8	20.5	11.6	11.6	11.6	11.6	9.2	9.2
32. TCx10^{-1}	221.5	81.5	61.5	54.6	85.0	63.9	50.8	42.5

4.27, 4.29, 4.32, and 4.34) with probabilistic coefficients. All the probabilistic constraints for each cutting-tool are supposed to be satisfied at the same probability level, i.e., $p_1 = p_2 = p_3$. Four probability levels, .5000, .8413, .9772, and .9887, are selected for the present study. The probability level of .5000 is equivalent to treat the constraint to be deterministic.

The following conclusions, common to both objective functions and the the two tool-replacement procedures, can be drawn :

1. The consideration of probabilistic tool-life has a significant effect on the machining parameters. The optimum cutting condition for the probability level .5000 differs highly from the other levels under consideration.

2. The cycle time decreases as the probability level increases. For example, in Table 9 the cycle time for the automatic Type-A goes down from .25 minutes to .05 minutes as the probability level changes from .5000 to .9987.

3. Except a few cases, speed and feed rate increases with probability level. For example, in Table 9 feed rate for first spindle for automatic type A goes up from .0079 ipr. to .0191 ipr. The same is true for speed.

4. It can be observed by glancing through the machining times, t_{ij}, in the Tables 9 - 12, that the variance in these time is more for automatic type - B compared to the type - A.

It is because of availability of more freedom in selection of machining parameters in Type - A.

5. The cycle time for a given probability level is less for the automatic type - B as compared to the type - A. The observation remains unexplainable.
6. The various tool-travel lengths for the operations to be performed on the job (Fig. 4) varies from .125 inches to 2.5 inches. The drilling operation on the 4th spindle has got the maximum machining length, i.e., 2.5 inches. Therefore, it can be expected that the 4th spindle would dictate the cycle time. The results compiled here are for 32 (4 probability levels x 2 types of automatics x 4 tables for different objective functions) sets of values. Out of the total 32 sets, it is observed that the 4th station is having the maximum machining time 24 times.
7. The results show that the machining parameters are insensitive to the capacity of the automatic. The Tables 9 and 10 are for automatic of 10 HP and 20 HP capacity respectively. It is observed that the values of machining parameters obtained for both the cases are the same. It is due to particular choice of job and cutting-force equation exponents.

The conclusions drawn above show that a higher productivity is possible in some cases with increased confidence in tool-life constraints. This also suggests that the current practices regarding selection of machining parameters based on deterministic tool-life values may yield a higher cost of production. However, an observation of opposite nature (a low

productivity with higher confidence level) is also possible with different set of data.

For the sake of clarity, the results of the two objective functions are discussed separately. The discussions of the results are given in the following sections.

7.2 Cycle Time Minimization

The Tables 9 and 10 depict the results for this objective function. The machining time is inversely proportional to the speed and feed rate, and therefore the minimization of cycle time is equivalent to maximizing speed and feed rate. The Table 9 shows that for high probability level (.9987), the speed for drilling operation on 4th spindle is very near to its maximum possible value (2400 rpm). A change in speed from 1269 rpm. to 2399 rpm. occurs for the drilling operation as the probability level goes up from .5000 to .9987. Both types of the automatics reveal similar results for feed rate and speed.

7.3 Total Production Cost Minimization

This objective function is studied under two types of tool-replacement procedures. The Tables 11 and 12 depict the results for unscheduled and scheduled tool-replacement procedure (2-A) respectively.

The magnitude of the various costs obtained are subject to time and cost parameters taken in this study.

However, it can be observed that the machine and labour cost is most important for the assumed input data. The second level of importance goes to the non-productive costs and obviously the least important is the tool-cost. An extensive sensitivity analysis on time and cost parameters is needed to draw any generalized conclusions regarding the relative importance of these three costs.

In the following sections the two tool-replacement procedures are discussed.

7.3.1 Unscheduled Tool-replacement Procedure

It is observed in Table 11 that a very low cycle time is possible in automatic type-B compared to type-A. For example, for the various probability levels under considerations, the cycle times in minutes for type-A are .32, .34, .33, and .31 while for type-B they are .31, .09, .06, and .06. The author is unable to explain this observation.

7.3.2 Scheduled Tool-replacement Procedure (2-A)

The results for this procedure are presented in Table 12. A factor 'C_{ij}' is defined in section 4.2.2 of Chapter 4. It is written as

$$C_{ij} = 1 - \frac{T_{ij} - q^{\sigma} T_{ij}}{SI}$$

where

T_{ij} - mean tool-life of i^{th} tool on j^{th} spindle,

$\sigma_{T_{ij}}$ - standard deviation of T_{ij} ,

SI - tool-replacement interval, and

$-2 \leqslant q \leqslant 2$

A full factorial experimental design was conducted for the factor q and the various probability levels under consideration. All integer values of q (-2 to 2) and the four probability levels (.5000, .8713, .9772, and .9987) were considered for the study. It was found that for a given value of probability level, the variation in q does not effect the optimal cutting conditions. This observation can be attributed to the selected mean values and the standard deviation of the tool life exponents.

The scheduled tool-replacement procedure is characterized by the duration of the period after which all the tools are replaced. It is noted that the number of schedules required to produce 21000 (20000 faultless components + 5% attrition allowance) components decreases from 34 to 6 as probability level is changed from .5000 to .8413. This is also to be noted that as probability level increases further, the drop in number of schedules is not that steep.

7.4 Comparison of the Two Tool-replacement Procedures

The choice among the two tool-replacement procedures under study can not be decided in general, since the relative merit depend upon the various time and cost parameters, and the mean and the variances of the exponents in tool-life equation.

However, the results obtained in this study reveal that the total cost of production is less in case of unscheduled tool-replacement compared to the scheduled tool-replacement procedure. This may be because of high tool life values resulting in less tool failures and also a high value (125 minutes) chosen for the time required to change all the tools simultaneously for scheduled tool-replacement increases total cost.

7.5 A Word about Solution Procedure

The solution procedure does not guarantee global optimum. Therefore, the results obtained here may not correspond to the global optimum. The optimum conditions can be improved by starting the algorithm with different initial solution and thus selecting the minimum of all minimums. This improvement also can not ensure to reach global optima but it will yield a better solution and also the behaviour of the objective function will be revealed.

The efficiency of the algorithm can also be improved by a proper selection of initial step size in Golden Search Section. The author has experienced that the initial value of step size affects greatly the total computer time required to reach the optimum solution.

7.6 General Conclusions

In this study the determination of optimum cutting conditions is integrated with tool-replacement procedures and they can be selected analytically. Furthermore, it

shows that the probabilistic consideration of tool-life effects the cutting conditions, cycle time and, the total cost of production significantly. It can be concluded that the decisions based on deterministic values of tool-life may yield a low productivity in some cases as observed in the present work.

As pointed out in Chapter VI, the data used for the analysis are primarily representative in nature. The involved arbitrariness in data prohibits one to make any generalizations.

CHAPTER VIII

SCOPE OF FUTURE WORK

8.1 Optimization Procedure

A solution methodology can be used by an industry if it is economical and easy to adopt it. Any methodology suggested can turn out to be economical in two ways; 1) the solution obtained by employing the suggested method brings appreciable reduction in cost of production and 2) the expenditures incurred in using the methodology should be less than the saving in cost of production. The reduction in cost of production can be achieved by employing the algorithm suggested in the present work. However, the efficiency of the package can be increased either by making improvements in the algorithm or by using a different method of solution. The interior penalty function approach, though recommended by many authors, has been found sluggish and time consuming in the present work. The author recommends to try another method, such as Rosen's Gradient Projection method (40), to solve the problem. The relative merit of the two methods can be evaluated and the better of the two should be selected.

8.2 Optimum Grouping of Operations

As mentioned in section 2.2 of Chapter II, the area of optimum grouping of the operations in a multi-tool case remains relatively unexplored. The problem also involves the redefining of operations and tool interference phenomena. The author feels

that the graphical technique, referred as Analysis of Interconnected Decision Areas (26) in the literature can be used to solve this problem.

8.3 Application of Group Technology

Another field open for further research is the application of the concepts of group technology in multi-spindle automatics. A lot of time is wasted in change-over of the production from one job to the other. The group technology concepts can be used to group the jobs in such a way so as minimum alterations are needed for the change-overs. Once such group is found out, the technique of Branch and Bound can be put to use to determine an optimum sequence of jobs to be manufactured.

8.4 Tool-life experiments for Form Tools

The literature available shows that the form tools have received very little attentation. An experimental research can be conducted whether the tool values for form tools differ significantly from single point tool or not.

8.5 Effect of Horsepower of the Machine Tool

In Chapter VII it was mentioned, on the basis of data taken for this study, that the capacity of machine tool in terms of its horsepower, does not affect the results. This observation does not seem to be convincing. Therefore, it may be worthwhile idea to investigate the affect of machine tool capacity using various sets of data.

REFERENCES

1. Armarego, E.J.A., and Russel, J.K., "Maximum Profit Rate as a Criterion for Selection of Machining Conditions", International Journal of Machine Tool Design and Research, Vol. 6, 1966, pp 15 - 23.

2. Armarego, E.J.A., and Russel, J.K., "The Maximum Profit Rate Criterion Applied to Single Pass Shaping and Milling Process", International Journal of Machine Tool Design and Research, Vol. 7, 1967, pp. 107 - 121.

3. Bartalucci, B., Bedini, B., and Lisini, G.G., "Economics in Machining : Optimization on Automatic Lathes," SME paper MR 70 - 235.

4. Berra, P. Bruce, and Barash, Moshe M., "Automated Planning and Optimization of the Metal Working Processes", Report No. 14, Purdue Laboratory of Applied Industrial Control, Purdue University, Lafayette, Indiana.

5. Bhattacharya, A., and Faria Gonzalez, R., "Regression Analysis for Predicting Surface Finish and its Application in the Determination of Optimum Machining Conditions", Journal of Engineering for Industry, Vol. 92, 1970, pp. 711.

6. Bracken, J., and McCormick, G.P., Selected Applications of Non-Linear Programming, John - Wiley & Sons., U.S.A., 1968, pp. 94 - 100.

7a. Brown, R.H., "On the Selection of Economical Machining Rates", The International Journal of Production Research, Vol. 1, No. 2, March, 1962, pp. 1 - 22.

7. Brewer, R.C., "On the Economics of Basic Turning Operation", Trans. A.S.M.E., Vol. 80, 1958, pp. 1479 - 89.

8. Colding, B.N., "A Three Dimensional Tool-Life Equation - Machining Economics", Journal of Engineering for Industry, August 1959, pp. 239 - 249.

9. Cook, N.H., Manufacturing Analysis, Addison - Wesley Publishing Company, Inc. Reading, Massachussetts, 1966.

10. Davidon, W., 'Variable Metric Method for Minimization', A.E.C. Res. Develop Rept. ANL - 5990 (Rev.), 1969.

11. Duncan, J. (1969), "Mathematical and Numerical Methods for Scheduling Cutting Tool," Ph.D. Dissertation Published by University Microfilms, University Microfilms Ltd., High Wycomb, England.

12. Ebert, R.J., and Hershauer, J.C., "Tool Replacement Policies : Variations in Cost, Tool Quality and Monitoring Rules", AIIE Trans. Vol. 2, 1970, No. 2, pp. 142 - 149.

13. Ermer, D.S., "A Bayesian Model for Machining Economics", Trans. of A.S.M.E., Journal of Engineering for Industry, 1970, p. 628.

14. Ermer, D.S., and Gonzalez - Faria, R., "An Analytical Sensitivity Study of the Optimum", A.S.M.E. Paper No. 67 - WA/Prod. - 20.

15. Ermer, D.S., and Morris, M.S., "A Treatment of Error of Estimation in Determining Optimum Machining Conditions", International Journal of Machine Tool Design and Research, Vol. 9 - 10, 1969 - 70, p 357 - 362.

16. Ermer, D.S., and Wu, S.M., "The Effect of Error on the Determination of the Optimum Metal Cutting Conditions", Trans. of A.S.M.E. - Journal of Engineering for Industry, Vol. 89 1967, p. 315 - 322.

17. Ernst, H. Merchant, W.E., and Krabachar, "Radio - Active Cutting - Tools for Rapid Tool - Life Testing", Semi-Annual Meeting A.S.M.E., Cincinnati, June 1952.

18. Fiacco, A.V., and McCormic, G.P., Non - Linear Programming; Sequential Unconstrained Minimization Techniques, New York, Wiley 1969.

19. Field, M., Williams, R., Zlatin, N., and Kronnenberg, M., "Computerized Determination and Analysis of Cost and Production Rates for Machining Operations : Part 1 - Turning", A.S.M.E. Paper No. 67 - WA/Prod - 18.

20. Field, M., Williams, R., Zlatin, N., and Kronenberg, M., "Computerized Determination and Analysis of Cost and Production Rates for Operations Part 2 - Milling Drilling, Reaming and Tapping", A.S.M.E. Paper No. 68-WA/Prod-17.

21. Fletcher, R., and Powell, M.J.D., "A Rapidly Convergent Descent Method for Minimization", The Computer Journal, Vol. 6, No. 2, July 1963, pp. 163 - 168.

22. Fox, R.L., Optimum Methods for Engineering Design, Reading, Mass. Addison - Wesley Publishing Company, 1971.

22. Gilbert, W.W., "Economics of Machining", Chapter in Machining Theory and Practice, American Society for Metals, 1950.

23. Gilman, A.M., "Optimum Conditions for Metal Cutting Machine Tools", Russian Engineering Journal, Vol. 48, 1968, No. 9, p. 59.

24. Goranskii, G.K., "Theory of Automation of Production Planning and of Tooling, Israel Program for Scientific Translations Jerusalem, 1967.

25. Iwata, K., Murotsu, Y., Iwatsubo, T., and Fujii, S., "A Probabilistic Approach to the Determination of the Optimum Cutting Conditions," Unpublished Paper, Kobe University, Kobe, Japan, May, 1971.

26. Jessop, F.H., Luckman, N.J., and Stringer, J., "Analysis of Interconnected Decision Areas - An Algorithm for Project Development", Nature 206, No. 597a, 1965, April 13, p. 118.

27. Jones, W.L., and Morgan, J.R., "Use of Decision Diagrams to Examine Tooling Problems", The International Journal of Production Research, Vol. 6, 1967, No. 2, pp. 123 - 134.

28. Krishna, C., and Berra, P.B., "Optimization of a Face Milling Process by Convex Programming, "A.S.M.E. Paper No. 71 - Prod 10, 1971.

29. Krishna, C., and Berra, P.B., "A System for Automated Planning and Optimization of a Face - Milling Process" - Sycracuse University, Unpublished paper.

30. Kronenberg, M. "Machining Science and Applications, Pergaman Press 1966.

31. Lasdon, L.S., and Waren, A., "Mathematical Programming for Optimal Design", (Photo - stat copy) Electro - Technology, Nov., 1967, pp. 53 - 70.

32. McCullough, E.M., "Economics of Multi Tool Lathe Operation", Journal of Engineering for Industry, 1963, pp. 402-405.

33. Merchant, M.E., "Progress and Problems in the Application of New Optimization Technology in Manufacturing", Annals of C.I.R.P. Vol. 16, pp. 151 - 161.

34. Okushima, K. and Fujii, S., "A Contribution to Determination of Tool - change Interval", Bulletine of JSME, Vol. 12, No. 52, 1969, pp 910 - 917.

35. Okushima, K., and Hitomi, K., "A Study of Economic Machining - An Analysis of the Maximum Profit Cutting Speed", International Journal of Production Research, Vol. 3, No. 1, 1964, pp. 73 - 78.

36. Opitz, H., and Weindahl, H.P., "Group Technology and Manufacturing Systems for Small and Medium Quantity Production, Int. J. Prod. Res., Vol. 9, No. 1, 1971, pp 181 - 203.

37. Pasko, N.I., "Mean Life of Multi - Tool Set Up", Russian Engineering Journal, Vol 49, No. 1, p. 63.

38. Rabinow, K.E., "Job - Shop Optimization", Submitted in Partial Fulfilment of the Requirement for the Degree of M.S. at the M.I.T., June, 1968.

39. Rajgopalan, R., "Cellular Production Systems - A Graph Theory Approach", Submitted in Partial Fulfilment of the Requirement for Degree of M.Tech. at the I.I.T. Kanpur, Aug.,1973.

40. Rosen, J.B., "The Gradient Projection Method for Non - Linear Programming; Part I", Journal of the Society of Mathematics, V. 9, No. 1, March, 1960, pp 181 - 217.

41. Scott, R.B., "Regenerative Shop Planning", American Machinist, Vol. 110, March 28, 1966, p 89 - 94.

42. Taha, A.H., "A Policy for Determining the Optimal Cycle Length for a Cutting Tool", The Journal of Industrial Engineering, Vol. 16, No. 3, 1966, pp 157 - 162.

43. Taylor, R.W., "On the Art of Cutting of Metals", Trans. A.S.M.E, Vol. 28, 1906, pp 310 - 350.

43a. Taylor, F.W., Proceedings of Institution of Engineers (London), pp 258 - 289, 1946.

44. Vulfson A., Deryabin, A.L., "Speeds and Feed Algorithm for Programme Controlled Lathe", Russian Engineering Journal, 1967, Vol. 47, No. 4,p 68.

45. Vyoskovoskii, E.S., "Reliability of Tools used in Semi - Automatic Lathes", Russian Engineering Journal, Vol. 4, No. 6, 1966, p 46.

46. Wager, J.G., Barash, M.M., "The Nature and Significance of the Distribution of High - speed - steel tool-life", Unpublished Doctoral Dissertation,Purdue University, 1967.

47. Weill, R., "Optimizing Machining Conditions in Numerical and Adaptive Control,"Proceedings of the 7th International Conference on Manufacturing Technology, Ann Arbor, Michigan, 1967, pp 569 - 580.

48. Wu, S.M., Ermer, D.S., "Maximum Profit as the Criterion of Determining the Optimum Cutting Conditions", Journal of Engineering for Industry, Nov. 1966, Vol. 88, pp 435-442.

49. Wu, S.M., "Tool-Life Testing by Response Surface Methodology; Part I and II", Journal of Engineering for Industry, May 1964, pp 105 - 110, 110 - 116.

50. Wu, S.M. and Tee, L.H., "Computerized Determination of Optimum Cutting Condition for a Fixed Demand", Advances in M.T.D.R, 1968, pt. 2, pp 1235 - 1253.

51. Zoutendijk, G., Method of Feasible Directions; a Study in Linear and Non - Linear Programming, Amsterdam, Elsevier, 1960.

APPENDIX A

DERIVATION OF METHOD OF CONVERTING CHANCE CONSTRAINTS INTO DETERMINISTIC CONSTRAINTS

Define random variables y_i's, as follows :

$$y_i = \sum_{j=1}^{n} a_{ij} \; x_j - b_i \qquad i = 1, 2, \dots , m \qquad \text{(A-1)}$$

where

a_{ij}'s are the technological coefficients, and b_i's are the constraint coefficients.

Since the sum of normal random variables is also a random variable, y_i is a normal random variable. The mean value and the standard deviation of y_i are given by

$$y_i = \sum_{j=1}^{n} \bar{a}_{ij} \; x_j - \bar{b}_i \qquad \text{(A-2)}$$

$$\sigma_{y_i} = \Bigg(\sum_{j,k=1}^{n} \rho_{a_{ij}\,a_{ik}} \, \sigma_{a_{ij}} \sigma_{a_{ik}} \; x_j \; x_k - \sum_{j=1}^{n} \rho_{a_{ij}\,b_i} \, \sigma_{a_{ij}} \sigma_{b_i} \; x_j + \sigma^2_{b_i} \Bigg)^{1/2} \qquad \text{(A-3)}$$

Then the chance constraint (4.16) is written as

$$\text{Prob } (y_i \geqslant 0) = \int_{-u}^{\infty} \frac{1}{\sqrt{2\pi}} \; e^{-t^2/2} \; dt \geqslant p_i$$

$$= \int_{\varphi'(p_i)}^{\infty} \frac{1}{\sqrt{2\pi}} \; e^{-t^2/2} \; dt$$

where

$$u = \bar{y}_i \,/\, \sigma_{y_i}$$

Hence we have

$$- \frac{\bar{y}_i}{\sigma_{y_i}} \leq \psi(p_i)$$

and thus

$$\bar{y}_i + \psi(p_i)\,\sigma_{y_i} \geqslant 0 \tag{A-4}$$

Substituting equations (A-2) and (A-3) into equation (A-4), the equation (4.17).

APPENDIX B

CHOICE OF INITIAL VALUE OF 'r'

To start the algorithm for unconstrained minimization a decision has to be taken about the initial values of 'r' and the factor c (the amount by which the value of 'r' is reduced at every step). The algorithm as such imposes no restriction whatsoever, on the value of 'r' and c. With large values of 'r' however, the number of computations required to reach the constrained minimum increases tremendously, while, small values of 'r' distort the function bringing in high eccentricity. The initial solution in the latter case has to be chosen sufficiently close to the actual minimum.

Fiacco and McCormick (18) have advanced two methods for the determination of an initial value of 'r_1', were

$$r_1 = f(\bar{x}_o)\, p(\bar{x}_o) \Big/ \left[p(\bar{x}_o) \right]^2 \qquad \text{(B-1)}$$

where

$\bar{x}_o$ is initial solution vector and

$$p(\bar{x}_o') = \sum_{i=1}^{m} 1 / g_i(\bar{x}_o) \qquad \text{(B-2)}$$

The equation (B-2) chooses a value of r_1 which makes the value of objective function and penalty term equal.

If $r < 0$, minimization of f (x) alone is carried out, without considering the penalty term. At every new point the value of r is checked by equation (B-1). As soon as $r > 0$, the minimization of $\phi(x, r)$ is started with gradual decrement in r as

has already been outlined.

When $r = 0$, it implies that the unconstrained minima has been attained.

However, if the starting point is close to the boundary the value of the penalty term is excessive, thereby yielding a very small value of r. In such cases, the critical constraint is arbitrarily assigned some minimum value. Thus

$$r_1 = f(\bar{x}_o) / \sum_{i=1}^{m} \frac{1}{\min(g_i(\bar{x}_o), g_{min})} \qquad \text{(B-2)}$$

When the constraints are normalised any arbitrary value can be assigned to g_{min}.

APPENDIX C

CALCULATION OF GRADIENT VECTOR

In order to calculate the direction used in Davidon - Fletcher - Powell's method, it is necessary to calculate the gradient vector of the function $f(x)$ during the course of each iteration. In the absence of exact or explicit formulae for the needed derivatives the gradient vector is approximated by the method of finite differences.

The technique is simply to use the well known formulae of differences to represent the derivatives. The simplest of these is the approximation

$$\frac{\partial f}{\partial x_j} = (f (X^j) - f (X) / \triangle x_j \qquad (C-1)$$

where

$$X^j = (x_1, x_2, \dots , x_j + \triangle x_j, \dots, x_n)$$

$$X = (x_1, x_2, \dots , x_j, \dots, x_n) \text{ , and}$$

$$\triangle x_j = \text{a small change in the variable } x_j .$$

The following scheme is followed to calculate $\triangle x_j$ to suit the problem studied in this work. The steps are :

1. Calculate $\triangle x_j = \epsilon x_j$ where ϵ is a sufficiently small chosen value.
2. Let $x_j' = x_j + \triangle x_j$.

3. If $\Delta x_j = 0$, modify Δx_j to δ, where δ is a very small number arbitrarily chosen.

4. The constraint set is calculated and the x_j' is checked for its feasibility. If no constraint is violated, gradient is calculated by equation (C-1) and letting $j = j + 1$ return to Step 1, otherwise go to Step 5.

5. If Δx_j is negative, modify Δx_j to $- \Delta x_j \cdot \epsilon$ and go to Step 4, otherwise $\Delta x_j = -\Delta x_j$ and go to Step 4.

The value of ϵ and δ should be chosen according to the function. In the present work ϵ and δ are taken to be as .1 and .01 respectively. Fox (22) suggests that one way of ascertaining whether the value of ϵ is proper or not is to compute $\frac{\delta f}{\delta x_j}$ as in equation (C-1) for one value of Δx_j, recompute it for a smaller value of it, and then compare the results. If they agree sufficiently then the approximation is probably sound, otherwise the values of Δx_j need adjustment. Fox has also suggested a 'central formula' to reduce the 'truncation error' involved in formula (C-1), but it increases the amount of computation by 100%. The 'central formula' is

$$\frac{\delta f}{\delta x_j} = \frac{f(X_j^+) - f(X_j^-)}{2 \Delta x_j} \qquad \text{(C-2)}$$

where

$$X_j^+ = (x_1, x_2, \ldots, x_j + \Delta x_j, \ldots, x_n)$$
$$X_j^- = (x_1, x_2, \ldots, x_j - \Delta x_j, \ldots, x_n)$$

However, to avoid excessive computational effort, the first method is chosen in the present study.

APPENDIX D

COMPUTER PROGRAMME LISTING

```
***************************************************************************

     L I S T I N G   O F  T H E S I S   P R O G R A M                    *
               OPTIMISATION IN MULTI- SPINDLE AUTOMATICS                  *
                    -A PROBABILISTIC APPROACH
***************************************************************************
   AMATR(1) -TYPE OF STEEL
   AMATR(2) -BHN OF RAW MATERIAL
   AMATR(3) - YIELD STRENGTH OF MATRIAL
   AMATR(4) - DIAMETRE OF THE RAW MATERIAL
   AMATR(5) - LENGTH OF BAR
   AMATR(6) - NO. OF SUCH BARS REQUIRED
   AMATR(7) -RAW MATERIAL COST RATE
   COMR(I) - COMPONENT REGUIREMENT MATRIX
   COMR(1) - NO. OF COMPONENTS REQUIRED
   COMR(2) - MATERIAL GRADE SPECIFICATION
   COMR(3) - COMPONENT TOTAL LENGTH
   COMR(4) - MATERIAL REQUIREMENT
   COMR(5) - ATTRITION NO.
   COMR(6) - DATE OF RECEIVING ORDER
   COMR(7) - DUE DATE
   COMR(8) - PRODUCTION STARTING TIME
   COMR(9) -- GROUP NO. TO WHICH IT BELONGS
   COMR(10) - FINISH TIME OF PRODUCTION
   COMR(11) -PRODUCTION PER UNIT
   COMR(12) - COST OF PRODUCTION
   COMD(I,J,K) - DENOTES K DATA CHARACTERISTIC ON JTH OPERATION ON ITH STN.
   COMD(I,J,1) - SURFACE PRODUCED
   COMD(I,J,2) - TYPE OF OPERATION
   COMD(I,J,3) - ACTUAL TRAVEL OF OPERATION
   COMD(I,J,4) -DEPTH OF CUT
   COMD(I,J,5) -TOOL USED
   COMD(I,J,6) - FEED INDEX
   COMD(I,J,7 ) - SPEED INDEX
   COMD(I,J,8) - MATERIAL REMOVING LENGTH
   COMD(I,J,9 ) - SECONDARY SURFACE OR PRIMARY SURFACE
   COMD(I,J,10)- WHETHER THE OPERATION REQUIRES DEPTH OF CUT
   COMD(I,J,11) - OPERATION IS DONE ON AXIAL SLIDE OR CROSS SLIDE
   COMD(I,J,12) - SURFACE RMS VALUE
   COMD(I,J,13) - WHETHER SURFACE COMES UNDER SMOOTH CATEGORY OR ROUGH CATEGO
   COMD(I,J,14) LENGTH TOLERANCE
   COMD(I,J,15) - DIAMETRE TOLERANCE
   COMD(I,J,16) - HMAX
   COMD(I,J,17) - SPEED
   COMD(I,J,18) - FEED
   COMD(I,J,19) - OLD DIAM.
   COMD(I,J,20) - NEW DIAMETRE
   COMD(I,J,21)- SPINDLE REVOLUTION NEEDED
   COMD(I,J,22)-TIME OF OPRATION
   CMC(I,J) - J TH CHARACTERISTICS FOR ITH M/C
   CMC(I,1) - NO. OF STATIONS
   CMC(U,2) - NO. OF AXIAL SLIDES
   CMC(I,3) - TYPE OF M/C
```

```
C       CMC(I,4) - DIFFERENT SPINDLE RPM OR SAME
C       CMC(I,5) MINIMUM M/C RPM
C       CMC(I,6) - MAXIMUM MACHINE RPM
C       CMC(I,7) - MINIMUM FEED
C       CMC(I,8) MAXIMUM FEED
C       CMC(I,9) - MINIMUM DEPTH OF CUT
C       CMC(I,10) MAXIMUM DEPTH OF CUT
C       CMC(I,11. HORSE POWER
C       CMC(I,12) NO. OF SPINDLE SPEEDS
C       CMC(I,13) NO. OF FEEDS
C       CMC(I,14) RAPID TRAVERSE SPEED
C       CMC(I,15) - POWER TRANSMISSION EFFICIENCY
C       CMC(I,16) SPECIFIED M/V PRODUCTION RATE
C       CMC((I,17) - MAXIMUM DIMENSION IN DIA. OF JOB WHIDH CAN BE HANDELED
C       CMC(I,18) MAXIMUM DIMENSION OF JOB IN LENGTH WHICH CAN BE HACDELED
C       CMC(I,19) - AXIAL MINIMUM FEED
C       CMC(I,2 ) MAXIMUM AXIAL FEED
C       TCM(I,J,K) - DENOTES KTH TOOL ON JTH OPERATION ON ITH STATION
C       TCM(I,1) - TYPE OF THE TOOL
C       TCM((I,J,2) NO. OF TEETH
C       TCM(I,J,3) ANY OTHER RELEVANT TOOL DIMENSION
C       TCM(I,J,4) DIAMETRE OF THE TOOL
C       TCM(I,J,5)  RAKE ANGLE
C       TCM(I,J,6) WEAR ALLOWED
C       TCM(I,J,7) TOOL TIP RADIUS
C       TCM(I,J,8) MAXIMUM FORCE ALLOWABLE
C       TCM(I,J,9) ROUGHING TOOL OR FINISHING TOOL
C       TCM(I,J,1 ) NO. OF REGRINDS PERMISSIBLE
C       TCM(I,J,11)  TOOL EDGE COST
C       TCM(I,J,12) - HORCE POWER CONSTANT
C       COST(I) - DENOTES COST PARAMETRES
C       COST(1) DIRECT LABOR COST/
C       COST(2) OVER HEAD COST
C       COST(3) TOOL GRINDING COST
C       COST(4) MACHINE SETTING COST
C       COST(5) - INTERMEDIATE ADJUSTMENT LABOR COST
C       TIMEC(I,J,K)  DENOTES KTH TIME VARIABLE FOR JTH TOOL ON ITH STATION
C       TIMEC(I,J,1) SETTING TIME
C       TIMEC(I,J,,2) GRINDING TIME
C       TIMEC(I,J,3) INTERMEDIATE ADJUSTMENT TIME
C       TIMEC(,J,4) MACHINING TIME
C       CYCLT - CYCLE TIME
C       TOOL(I,J,K) KTH PARAMETRE OF JTH TOOL
C       TOOL(I,J,1) AVERAGE FEED EXPONENTS
C       TOOL(I,J,2) ST. DEV. OF FEED EXPONENT
C       TOOL(I,J,3) AVERAGE DEPTH OF CUT EXPONENT
C       TOOL(I,J,4) ST. DEV. OF DEPTH OF CUT EXPONENT
C       TOOL(I,J,5) AVERAGE SPEED EXPONENT
C       TOOL(I,J,6) - ST. DEV. OF SPEED EXPONENT
C       TOOL(I,J,7) - AVERAGE VALUE OF CONSTANT IN TOOL LIFE EQUATION
C       TOOL(I,J,8) ST. DEV. OF CONSTANT IN TOOL,LIFE EQUATION
C       FORCE(I,J,K) - DENOTES KTH PARAMETRE OF JTH TOOL ON ITH STATION
C       FORCE(I,J,1) AVE. VALUE OF FEED EXPONENT IN FORCE EQ.
```

```
C      FORCE(I,J,2) ST. DEV. FEED EXPONENT
C     FORCE(I,J,5) AVE. VALUE OF SPEED EXPONENT
C     FORCE(I,J,6) ST. DEV. OF SPEED EXPONENT
C     FORCE(I,J,7) - AVE VALUE OF CONSTANT IN FORCE EQ.
C     INDEXM - INDEX FOR MACHINE , WHETHR THE MACINE IS MULTI SINGLE TOOL
C     INDEXF - INDEX FOR OBJECTIVE FUNCTION
C     INDEXO - INDEX FOR OPERATION TYPE
C     INDEX - THIS INDDEX IS FOR SULUTION TO BE CARRIED TO THE NEXT PROBLEM
       COMMON/SERCH2/FUNVAL
      COMMON/REPT/PO,P,TP,ITNCH,TEC,TGCH,GCH,SETC,CIDLM,GRC,STEC,SGRC,
     1SSETC,SCIDLM,STGCH,ABC,NTEC,NGC,NGR,COSTP,SGC6,NTNCH,ASYLT,FACTOR
     1,NS
      DIMENSION PO(10),P(10),ITNCH(10),TEC(10),TGCH(10),GCH(10),SETC(10)
     1,CIDLM(10),GRC(10),STEC(10),SGRC(10),SSETC(10),SCIDLM(10),STGCH(10)
     1),ABC(10),NTEC(10),NGC(10),NGR(10)
        COMMON/LIFE/TLIF,TVARI,FVARI,NTA
      COMMON/STEP/ETA
      COMMON/TOR/FC2(10),G11(10),G22(10)
      COMMON/TOTAL/NCONST,NOBJ,NTFN
      COMMON/PRO/PROB
      COMMON/MINI/R
      COMMON/CYC/CYCLT,AMAXO
      COMMON/INEQLT/CON,SATSFY
      COMMON/MANE/N,NIN
      DIMENSION NSPO(10),AMATR(10),COMR(10),COMD(8,6,25),CMC(5,20),TCM(8
     1,6,25),COST(10),TIMEC(8,6,10),TOOL(8,6,15),FORCE(8,6,15),PROB(10),
     1T(10,10),TC(10,10),NJ(3,8,5),CON(100),HP1(10,10),HP2(10,10)
       DIMENSION X(50)
      DIMENSION TLIF(10),TVARI(10),FVARI(10)
       DIMENSION Y(50)
      COMMON/INVAR/INXE,INYE
      COMMON/DIV/C1
      COMMON/MATV/AMATR,COMR,COMD,CMC,TCM,COST,TIMEC,TOOL,FORCE,NSPO
      COMMON/NVAR/NAMAT,NCOMR,NSTN,NOPT,NCOMD,NCMC,NTCM,NCOST,NTIME,NTP,
     1NFP,NMACIN,NSPNDL,NAXIL,NCROSS,NPROB,MN,NSTION,NOP
      COMMON/IND/INDEX,INDEXF,INDEXM
      COMMON/OUT/Y,IOY,IOX,NJ,G1,G2,FC,HP1,HP2,T,TC
      DATACOSTP/.50/
      DATA SGC6/50./
      INDEXO=4
      ETA=.25
      INDEXF=3
      READ 1,INXE,INYE
      PRINT1,INXE,INYE
      READ 1,NAMAT,NCOMR,NSTN,NOPT,NCOMD,NCMC,NTCM,NCOST,NTIME,NTP,NFP,
     1NMACIN ,NSPNDL,NAXIL,NCROSS,NPROB
      PRINT 1,NAMAT,NCOMR,NSTN,NOPT,NCOMD,NCMC,NTCM,NCOST,NTIME,NTP,
     1NFP,NMACIN,NSPNDL,NAXIL,NCROSS,NPROB
      READ 3,(AMATR(I),I=1,NAMAT)
      PRINT 3,(AMATR(I),I=1,NAMAT)
      READ 4,(COMR(I),I=1,NCOMR)
      COMR(1)=COMR(1)*2.
      PRINT 4,(COMR(I),I=1,NCOMR)
```

```
      READ 2,(PROB(I),I=1,NPROB)
       PRINT 2,(PROB(I),I=1,NPROB)
      READ 5,(NSPD(I),I=1,NSPNDL)
      PRINT 5,(NSPD(I),I=1,NSPNDL)
5     FORMAT(16I5)
      DO6I=1,NSPMDL
      NN=NSPC(I)
      DO6J=1,NN
      READ 7,(COMD(I,J,K),K=1,NCOMD)
      PRINT 7,(COMD(I,J,K),K=1,NCOMD)
      IF(COMD(I,J,4).EQ.0.) GO TO 6
      COMD(I,J,4)=ALOG(COMD(I,J,4))
6     CONTINUE
      DO8I=1,NMACIN
      READ 9,(CMC(I,K),K=1,NCMC)
      PRINT 9,(CMC(I,K),K=1,NCMC)
8     CONTINUE
      DO10I=1,NSPNDL
      NN=NSPC(I)
      DO10J=1,NN
      READ 11,(TCM(I,J,K),K=1,NTCM)
      TCM(I,J,3)=TCM(I,J,8)/2.
      PRINT 11,(TCM(I,J,K),K=1,NTCM)
10    CONTINUE
      READ 12,(COST(I),I=1,NCOST)
      PRINT 12,(COST(I),I=1,NCOST)
      DO13I=1,NSPNDL
      NN=NSPC(I)
      DO13J=1,NN
      READ 14,(TIMEC(I,J,K),K=1,NTIME)
      PRINT 14,(TIMEC(I,J,K),K=1,NTIME)
13     CONTINUE
       DO17I=1,NSPNDL
      NN=NSPC(I)
      DO17J=1,NN
      READ 14,(TOOL(I,J,K) ,K=1,NTP)
      PRINT 14,(TOOL(I,J,K),K=1,NTP)
17     CONTINUE
      DO19I=1,NSPNDL
      NN=NSPC(I)
      DO19J=1,NN
      READ 14,(FORCE(I,J,K),K=1,NFP)
      PRINT 14,(FORCE(I,J,K),K=1,NFP)
19     CONTINUE
1     FORMAT(16I5)
2     FORMAT(10F6.2)
3     FORMAT(8F10.2)
4     FORMAT(2F10.0,/15F5.1)
7     FORMAT(8F10.3)
9     FORMAT(5F7.1,4F10.4,F5.1,/8F7.1,2F8.4)
11    FORMAT(10F8.3)
12    FORMAT(8F10.3)
14    FORMAT(8F10.3)
```

```
      GO TO (30,40,50,60,60,60),INDEXO
30     NSTION=1
90    INDEXM=1
      INDEXF=1
      TCOST=0.
      TTIME=0.
      NOPT=NSPO(NSPNDL)
      DO31IN=1,NOPT
      NOP=IN
      CALL CCNSTR(X)
      PRINT 35,NIN,N
      CALL MINIMA(X)
       PRINT 33,INDEXO
       PRINT 32,COST(10),IN,NSTION,T(NSPNDL,IN)
      TCOST=TCOST+COST(10)
      TTIME=TTIME+T(NSPNDL,IN)
31     CONTINUE
      TIMEPJ=TTIME/COMR(1)
      COSTPJ=TCOST/COMR(1)
      STOP
40    NSTION=1
      GO TO 90
50    NSTION=1
      INDEXM=2
      TCOST=0.
      TTIME=0.
      INDEXF=1
      DO34I=1,NSPNDL
      NN=NSPC(I)
      DO34J=1,NN
      CALL CCNSTR(X)
      PRINT 35,NIN,N
      CALL MINIMA(X)
      PRINT 33,INDEXO
      PRINT 32,COST(10),J,I,T(I,J)
       TCOST=TCOST+COST(10)
      TTIME=TTIME+T(I,J)
34     CONTINUE
      TIMEPJ=TTIME/COMR(1)
      COSTPJ=TCOST/COMR(1)
      STOP
60    DO53MN=1,NMACIN
      INDEX=1
      NSTION=CMC(MN,1)
      FACTOR=5.
      NCONST=0
       NOBJ=0
      NTFN=0
      CMC(MN,6)=2400.                                                        CARE
       IF(CMC(MN,3).EQ.0.) CMC(MN,11)=CMC(MN,11)/FLOAT(NSTION)
      INDEXM=2
      PRINT 33,INDEXO
      CALL CCNSTR(X)
```

```
      PRINT 35,NIN,N
      SUM=0.
      DO89IN=1,NIN
89    SUM=SUM+1./CON(IN)
      R=FUNVAL/SUM
      C =4.
      CALL MINIMA(X)
      DO36I=1,NSTION
      NN=NSPC(I)
       DO37J= ,NN
      IF(COME(I,J,9).EQ. .) GO TO 37
      PRINT 32,COST( ),J,I,T(I,J)
37    CONTINUE
36    CONTINUE
      PRINT 82,NCONST,NOBJ,NTFN
82    FORMAT(1X,5I1 )
      IF(INDEXF.EQ.2) GO TO 501
      IF(INDEXF.EQ.3) GO TO 502
       PRINT 505,CYCLT
       DO5 3II=1,NTA
503    PRINT 505,TLIF(II),TVARI(II),FVARI(II),FC2(II),G11(II),G22(II)
      GO TO 53
501    PRINT 505,ASYLT,CYCLT
504    DO506II=1,NTA
506    PRINT 505,TLIF(II),TVARI(II),FVARI(II),FC2(II),G11(II),G22(II)
      DO5 7II=1,NTA
507   PRINT 5 5,TEC(II),TGCH(II),SETC(II),GRC(II),CIDLM(II),ABC(II),
     1ITNCH(II),NTEC(II),NGR(II)
      IF(INDEXF.EQ.3) GO TO 509
      GO TO 53
502     GO TC 501
509     DO508II=1,NTA
508     PRINT 505,STEC(II),SGRC(II),STGCH(II),SSETC(II),SCIDLM(II)
        PRINT 5000,NS
5000  FORMAT(1X,5I10)
505    FORMAT(1X,6F12.2,5I10)
53    CONTINUE
57    CONTINUE
33    FORMAT(30X,*OPERATION INDEX=*,I5)
32    FORMAT(1X,*TOTAL COST=*,F10.2,/1X,*TIME OF OPERATION*,I5,2X,*ON*,I
     15,2X,*STATION*,F10.2)
35    FORMAT(10X,*NO. OF CONSTRAINTS=*,I5,/10X,*NO. OF VARIABLES=*,I5)
      STOP
      END


      SUBROUTINE INTAL(X,Y)
C
C     THIS SUBROUTINE INITIALIZE THE VARIABLES
C
      COMMON/INVAR/INXE,INYE
```

```
      DIMENSION X(5[illegible])
      DIMENSION Y(50)
      X(1)=.0[illegible]6
       X(2)=.[illegible]48
      X(3)=.0[illegible]7
      X(4)=.0007
       X(5)=.[illegible]
      Y(1)=30[illegible].
      Y(2)=60[illegible].
      Y(3)=600.
      Y(4)=60[illegible].
      Y(5)=60[illegible].
      Y(6)=60[illegible].
      DO1I=1,INXE
1      X(I)=ALOG(X(I))
      DO2I=1,INYE
2      Y(I)=ALOG(Y(I))
      RETURN
      END



      SUBROUTINE OBJFN(TCOST)
C
C      TIS SUBROUTINE EVALUARES THE TOTAL COST FOR TWO SCHEDULING POLICIES
C
      COMMON/REPT/PO,P,TP,ITNCH,TEC,TGCH,GCH,SETC,CIDLM,GRC,STEC,SGRC,
     1SSETC,SCIDLM,STGCH,ABC,NTEC,NGC,NGR,COSTP,SGC6,NTNCH,ASYLT,FACTOR
     1,NS
      COMMON/LIFE/TLIF,TVARI,FVARI,NTA
       DIMENSION TLIF(10),TVARI(10),FVARI(10),FXX(30),PO(10),P(10),ITNCH(10),
     1(10),TEC(10),TGCH(10),GCH(10),SETC(10),CIDLM(10),GRC(10),STEC(10),
     1SGRC(10),SSETC(10),SCIDLM(10),STGCH(10),ABC(10),NTEC(10),NGC(10),
     1NGR(10)
      COMMON/INVAR/INXE,INYE
      COMMON/CYC/CYCLT,AMAXO
      COMMON/MANE/N,NIN
      COMMON/MATV/AMATR,COMR,COMD,CMC,TCM,COST,TIMEC,TOOL,FORCE,NSPO
      COMMON/NVAR/NAMAT,NCOMR,NSTN,NOPT,NCOMD,NCMC,NTCM,NCOST,NTIME,NTP,
     1NFP,NMACIN,NSPNDL,NAXIL,NCROSS,NPROB,MN,NSTION,NOP
      COMMON/IND/INDEX,INDEXF,INDEXM
      COMMON/OUT/Y,IOY,IOX,NJ,G1,G2,FC,HP1,HP2,T,TC
      COMMON/INEQLT/CON,SATSFY
      DIMENSION NSPO(10),AMATR(10),COMR(10),COMD(8,6,25),CMC(5,20),
     1TCM(8,6,25),COST(10),TIMEC(8,6,10),TOOL(8,6,15),FORCE(8,6,15),
     2PROB(10),CON(100),Y(50),HP1(10,10),HP2(10,10),X(50),T(10,10),TC(10,10)
     3,10),NJ(3,8,5)
      ASYL=0.
```

```
      KN=0
      DO1I=1,NSTION
      NN=NSPC(I)
      DO2J=1,NN
       IF(COMD(I,J,9).EQ.0.) GO TO 2
      KN=KN+1
      ASYL=ASYL+1./TLIF(KN)
2     CONTINUE
1     CONTINUE
      ASYLT=1./ASYL
      TPROT=1.1*CYCLT*COMR(1)
      NTNCH=TPROT/ASYLT+1.
      PROC=TPROT*COST(1)
      GO TO (30,10,20),INDEXF
10    KN=0
      DO4I=1,NSTION
      NN=NSPC(I)
      DO5J=1,NN
      IF(COMD(I,J,9).EQ.0.) GO TO 5
      KN=KN+1
      TPROT=T(I,J)*COMR(1)
      ITNCH(KN)=TPROT/TLIF(KN)+1.-1.
      AIND=1.
      ABC(KN)=TPROT/TLIF(KN)
      IF(ABC(KN).LT.TCM(I,J,10))AIND=0.
      NTEC(KN)=ABC(KN)/(TCM(I,J,10)+1.)+1.
      TEC(KN)=FLOAT(NTEC(KN))*TCM(I,J,11)
      NGR(KN)=ABC(KN)-TCM(I,J,10)+1.-1.
      GRC(KN)=FLOAT(NGR(KN))*COST(3)*TIMEC(I,J,2)*AIND
      SETC(KN)=FLOAT(ITNCH(KN))*TIMEC(I,J,3)*COST(5)
      CIDLM(KN)=(COSTP+COST(7))*(TIMEC(I,J,3)/CYCLT)*FLOAT(ITNCH(KN))
      TGCH(KN)=0.
5     CONTINUE
4     CONTINUE
      TCOST=PROC
      DO6I=1,KN
      TCOST=TCOST+TEC(I)+TGCH(I)+GRC(I)+SETC(I)+CIDLM(I)
6     CONTINUE
      RETURN
20    NS=1.1*COMR(1)/EXP(Y(1))
      KN=0
      DO7I=1,NSTION
      NN=NSPC(I)
      DO8J=1,NN
      IF(COMD(I,J,9).EQ.0.) GO TO 8
      KN=KN+1
       PO(KN)=1.-(TLIF(KN)-P(KN)*TVARI(KN)/FACTOR)/(Y(1)*CYCLT)
      IF(PO(KN).LT.0.) PO(KN)=0.
      IF(PO(KN).GT.1.) PO(KN)=1.
      ITNCH(KN)=PO(KN)*FLOAT(NS)+1.-1.
      AIND=1.
      IF(FLOAT(ITNCH(KN)).LE.TCM(I,J,10))AIND=0.
      NTEC(KN)=(PO(KN)*FLOAT(NS))/(TCM(I,J,10)+1.)+1.
```

```
      NGR(KN)=FLOAT(ITNCH(KN))-TCM(I,J,10)+1.-1.
      TEC(KN)=FLOAT(NTEC(KN))*TCM(I,J,11)
      STEC(KN)=FLOAT(NS)/(TCM(I,J,10)+1.)*TCM(I,J,11)
      STGCH(KN)=0.
      GRC(KN)=FLOAT(NGR(KN))*COST(3)*TIMEC(I,J,2)*AIND
      SGRC(KN)=(FLOAT(NS)-TCM(I,J,10))*COST(3)*TIMEC(I,J,2)*AIND
      SETC(KN)=FLOAT(ITNCH(KN))*TIMEC(I,J,3)*COST(5)
      SSETC(KN)=FLOAT(NS)*TIMEC(I,J,3)*COST(5)
      CIDLM(KN)=(TIMEC(I,J,3)/CYCLT)*FLOAT(ITNCH(KN))*(COSTP+COST(7))
      SCIDLM(KN)=(COSTP+COST(7))*FLOAT(NS)*(TIMEC(I,J,3)/CYCLT)
      TGCH(KN)=0.
 8    CONTINUE
 7    CONTINUE
      SCOST=SGC6*FLOAT(NS)*COST(3)
      TUCST=0.
      DO9I=1,KN
      TUCST=TUCST+TEC(I)+TGCH(I)+GRC(I)+SETC(I)+CIDLM(I)
 9    CONTINUE
      TSCST=SCOST
      DO11I=1,KN
      TSCST=TSCST+STEC(I)+STGCH(I)+SGRC(I)+SSETC(I)+SCIDLM(I)
 11   CONTINUE
      TCOST=TSCST+TUCST+PROC
 30    RETURN
      END



       SUBROUTINE OUTPUT(X)
C
C     THIS SUBROUTINE PRINTS VALUES OF DIFFERENT VARIABLES
C
      COMMON/INVAR/INXE,INYE
      COMMON/MANE/N,NIN
      COMMON/MATV/AMATR,COMR,COMD,CMC,TCM,COST,TIMEC,TOOL,FORCE,NSPD
      COMMON/NVAR/NAMAT,NCOMR,NSTN,NOPT,NCOMD,NCMC,NTCM,NCOST,NTIME,
     1NTP,NFF,NMACIN,NSPNDL,NAXIL,NCROSS,NPROB,MN,NSTION,NOP
      COMMON/IND/INDEX,INDEXF,INDEXM
      COMMON/OUT/Y,IOY,IOX,NJ,G1,G2,FC,HP1,HP2,T,TC
      DIMENSION NSPD(10),AMATR(10),COMR(10),COMD(8,6,25),CMC(5,20),
     1TCM(8,6,25),COST(10),TIMEC(8,6,10),TOOL(8,6,15),FORCE(8,6,15),
     2PROB(10),CON(100),Y(50),HP1(10,10),HP2(10,10),X(50),T(10,10),TC(10,
     3,10),NJ(3,8,5)
      DIMENSION XX(50),YZ(50)
      GO TO (10,20),INDEXM
 20    DO1I=1,INXE
 1    XX(I)=EXP(X(I))
      IDX=INXE+1
      DO2I=IDX,N
      IY=I-INXE
 2    XX(I)=EXP(Y(IY))
      PRINT 3,(XX(I),I=1,N)
```

```
3     FORMAT(1X,*OUTPUT*,5F15.4)
10    RETURN
      END


       SUBROUTINE CONSTR(X)
C*****************************************************************************
C     THIS SUBROUTINE IS TO GENERATE CONSTRAINTS FOR A MULTI-SPINDLE AUTOMATOC
C                 MACHINE HAVING INDEPENDENT DRIVES FOR EACH SPINDLE
C*****************************************************************************
      DIMENSION NSPO(10),AMATR(10),COMR(10),COMD(8,6,25),CMC(5,20),TCM(8
     1,6,25),COST(10),TIMEC(8,6,10),TOOL(8,6,15),FORCE(8,6,15),PROB(10)
      DIMENSION CON(100),Y(50),HP1(10,10),HP2(10,10),X(50)
      DIMENSION T(10,10),TC(10,10)
      DIMENSION NJ(3,8,5)
      COMMON/TOTAL/NCONST,NOBJ,NTFN
      COMMON/SERCH2/FUNVAL
      COMMON/CYC/CYCLT,AMAXO
      COMMON/PRO/PROB
      COMMON/MATV/AMATR,COMR,COMD,CMC,TCM,COST,TIMEC,TOOL,FORCE,NSPO
      COMMON/NVAR/NAMAT,NCOMR,NSTN,NOPT,NCOMD,NCMC,NTCM,NCOST,NTIME,NTP,
     1NFP,NMACIN,NSPNDL,NAXIL,NCROSS,NPROB,MN,NSTION,NOP
      COMMON/IND/INDEX,INDEXF,INDEXM
       COMMON/OUT/Y,IOY,IOX,NJ,G1,G2,FC,HP1,HP2,T,TC
      COMMON/SUB4/SIGMA
      COMMON/MANE/N,NIN
      COMMON/INEQLT/CON,SATSFY
      COMMON/INVAR/INXE,INYE
       DIMENSION PROBHP(5)
        COMMON/LIFE/TLIF,TVARI,FVARI,NTA
        COMMON/TOR/FC2(10),G11(10),G22(10)
      DIMENSION TLIF(10),TVARI(10),FVARI(10)
      NTA=0
      IF(INDEX.EQ.0.AND.INDEXM.EQ.2) GO TO 25
      IF(INDEX.EQ.1.AND.INDEXM.EQ.1) GO TO 31
      IF(INDEX.EQ.0.AND.INDEXM.EQ.1) GO TO 400
      CALL INTAL(X,Y)
C     INITIALISATION OF X AND Y
      INDEX=0
      GO TO 27
31     CALL INIT(X)
      INDEX=0
      GO TO 27
25     DO28I=1,IOY
      IDX=IOX+I
        Y(I)=X(IDX)
28     CONTINUE
27     IF(INDEXM.EQ.1) GO TO 400
       IOAXIL=0
      NCONST=NCONST+1
      NCON=0
      NSPNDL=1
```

```
12    I=NSPNCL
      NN=NSPC(I)
      PROBHP(1)=0.
      SHP=0.
      PROBHP(2)=1.
      PROBHP(4)=1.
      PROBHP(3)=0.
      DO2J=1,NN
      IF(COMD(I,J,9).EQ.1.) GO TO 2
      IF(COMD(I,J,11).EQ.0.) GO TO 3
      IF(COMD(I,J,6).GT.1.) GO TO 4
      NTA=NTA+1
      INX=I+1-IOAXIL
      PHIP1=PROB(1)
      PHIP2=PROB(2)
      CHP=TCM(I,J,12)*(10.**(-5))
      NJ(1,I,J)=INX
      IOX=INX
      NCON=NCON+1
C     MAXIMUM FEED CONSTRAINT
      CON(NCON)=ALOG(CMC(MN,8))-X(INX)
      NCON=NCON+1
C     MINIMUM FEED CONSTRAINT
      CON(NCON)=-ALOG(CMC(MN,7))+X(INX)
      GO TO 5
 4    INX=IOX
      NJ(1,I,J)=INX
      NTA=NTA+1
 5    IF(COMD(I,J,7).GT.1.) GO TO 6
      INY=I+1
      NJ(2,I,J)=INY
      IOY=INY
      NCON=NCON+1
C     MAXIMUM SPEED CONSTRAINT
      CON(NCON)=ALOG(CMC(MN,6))-Y(INY)
      NCON=NCON+1
C     MINIMUM SPEED CONSTRAINY
        CON(NCON)=-ALOG(CMC(MN,5))+Y(INY)
      GO TO 8
 6    INY=IOY
      NJ(2,I,J)=INY
      GO TO 8
 3    INX=1
      NTA=NTA+1
      IOAXIL=IOAXIL+1
      NCON=NCON+1
C     MAXIMUM FEED CONSTRAINT
      CON(NCON)=ALOG(CMC(MN,20))-X(1)
       NJ(1,I,J)=1
      NCON=NCON+1
C     MINIMUM FEED CONSTRAINT
      CON(NCON)=-ALOG(CMC(MN,19))+X(1)
      IF(COMD(I,J,7).GT.1.) GO TO 8
```

```
      INY=I+1
      NJ(2,I,J)=INY
      IOY=INY
      NCON=NCON+1
C     MAXIMUM SPEED CONSTRAINT
      CON(NCON)=ALOG(CMC(MN,6))-Y(INY)
      NCON=NCON+1
C     MINIMUM SPEED CONSTRAINY
      CON(NCON)=-ALOG(CMC(MN,5))+Y(INY)
 8    NCON=NCON+1
      G1=TOOL(I,J,8)**2+(TOOL(I,J,6)*Y(INY))**2+(TOOL(I,J,2)*X(INX))**2+(TOOL(I,
     1(TOOL(I,J,4)*COMD(I,J,4))**2
      G1=(TOOL(I,J,8)*ALOG(10.))**2-TOOL(I,J,8)**2+G1
C     TOOL LIFE CONSTRAINT
      CON(NCON)=ALOG(10.)*TOOL(I,J,7)+(1.+TOOL(I,J,1))*X(INX)+(1.+TOOL(I
     1,J,5))*Y(INY)+TOOL(I,J,3)*COMD(I,J,4)-Y(1)+PHIP1*(SQRT(G1))
      TLIFE=CON(NCON)-PHIP1*(SQRT(G1))
      TLIFE=TLIFE+Y(1)
      CON(NCON)=CON(NCON)-ALOG(COMD(I,J,8))
      TLIF(NTA)=EXP(TLIFE)
      G2=FORCE(I,J,8)**2+(FORCE(I,J,6)*Y(INY))**2+(FORCE(I,J,2)*X(INX))**2
     1*2+(FORCE(I,J,4)*COMD(I,J,4))**2
      G2=G2-FORCE(I,J,8)**2+(FORCE(I,J,8)*ALOG(10.))**2
      NCON=NCON+1
C     MAXIMUM FORCE CONSTRAINT
      CON(NCON)=ALOG(TCM(I,J,8))-ALOG(10.)*FORCE(I,J,7)-FORCE(I,J,5)*Y(IN
     1NY)-FORCE(I,J,1)*X(INX)-FORCE(I,J,3)*COMD(I,J,4)+PHIP2*(SQRT(G2))
      FC=(10.**FORCE(I,J,7))*(EXP(Y(INY))**FORCE(I,J,5))*(EXP(X(INX))**
     1FORCE(I,J,1))*(EXP(COMD(I,J,4))**FORCE(I,J,3))
       HP1(I,J)=CHP*FC*EXP(Y (INY))*(COMD(I,J,19)+COMD(I,J,20))/2.
      TVARI(NTA)=TLIF(NTA)*SQRT(G1)
      FVARI(NTA)=FC*SQRT(G2)
      G22(NTA)=G2
      G11(NTA)=G1
      FC2(NTA)=FC
      NCON=NCON+1
C     SURFACE ACCURACY CONSTRAINT
      IF(COMD(I,J,13).EQ.0.) CON(NCON)=-X(1)+ALOG(2.*COMD(I,J,16)
     1/(SIN(2.*TCM(I,J,5))))
       IF(COMD(I,J,13).EQ.1.) CON(NCON)=2.*X(1)+ALOG(COMD(I,J,16)*8.
     1*TCM(I,J,M))
      PROBHP(1)=PROBHP(1)+ALOG(10.)*HP1(I,J)
      PROBHP(1)=(PROBHP(1)*FORCE(I,J,8))**2
      SHP=SHP+PROBHP(1)*PHIP2
      K=NJ(2,I,J)
      PROBHP(2)=PROBHP(2)+Y(K)*HP1(I,J)
      PROBHP(2)=(PROBHP(2)*FORCE(I,J,6))**2
       SHP=SHP+PROBHP(2)*PHIP2
       K=NJ(1,I,J)
      PROBHP(3)=PROBHP(3)+X(K)*HP1(I,J)
      PROBHP(3)=(PROBHP(3)*FORCE(I,J,2))**2
      SHP=SHP+PROBHP(3)*PHIP2
       PROBHP(4)=PROBHP(4)+ COMD(I,J,4)*HP1(I,J)
```

```
      PROBHP(4)=(PROBHP(4)*FCRCE(I,J,4))**2
      SHP=SHP+PROBHP(4)*PHIP2
      SHP=SQRT(SHP)*SQRT(PHIP2)
      HP2(I,J)=SHP
 2    CONTINUE
      N=NSPO(I)
      DO1 J=1,N
      IF(COMD(I,J,9).EQ. .) GO TO 10
      NCON=NCON+1
        CON(NCON)=CMC(MN,11)-HP1(I,J)+HP2(I,J)
C     HORCE POWER CONSTRAINT
 10    CONTINUE
      NSPNDL=NSPNDL+1
      IF(NSPNDL.GT.NSTION) GC TO 11
      GO TO 12
 11   DO26I=1,IOY
      IDX=IOX+I
      X(IDX)=Y(I)
 26   CONTINUE
       FUNVAL=FUNXON(X)
      DO45I=1,NSTION
      NN=NSPC(I)
      DO46J=1,NN
       IF(COMD(I,J,9).EQ.0.) GO TO 46
      NCON=NCON+1
       CON(NCON)=CYCLT-T(I,J)+.01
 46   CONTINUE
 45    CONTINUE
      NCON=NCON+1
C     NO. OF COMPONENTS CONSTRAINT
       CON(NCON)=ALOG(COMR(1))-Y(1)
      NIN=NCCN
      N=IOY+IOX
      RETURN
 400   I=NSTION
      J=NOP
      PHIP1=PROB(1)
      PHIP2=PROB(2)
      NCON=0
      MN=5
      IF(COMD(I,J,9).EQ.0.) RETURN
      CHP=TCM(I,J,12)*(10.**(-6))
      NCON=NCON+1
C     MAXIMUM FEED CONSTRAINT
      CON(NCON)=ALOG(CMC(MN,8))-X(1)
      NCON=NCON+1
C     MINIMUM FEED CONSTRAINT
      CON(NCON)=-ALOG(CMC(MN,7))+X(1)
      NCON=NCON+1
C     MAXIMUM SPEED CONSTRAINT
      CON(NCCN)=ALOG(CMC(MN,6))-X(2)
      NCON=NCON+1
C     MINIMUM SPEED CONSTRAINY
```

```
      CON(NCON)=-ALOG(CMC(MN,5))+X(2)
      NCON=NCON+1
C     MAXIMUM DEPTH OF CUT CONSTRAINT
      CON(NCON)=ALOG(CMC(MN,10))-X(3)
      NCON=NCON+1
C     MINIMUM DEPTH OF CUT CONSTRAINT
      CON(NCON)=-ALOG(CMC(MN,9))+X(3)
      NCON=NCON+1
      G1=TOOL(I,J,8)**2+(TOOL(I,J,6)*X(2))**2+(TOOL(I,J,2)*X(1))**2
     1+(TOOL(I,J,4)*X(3))**2
            G2=FORCE(I,J,8)+(FORCE(I,J,6)*X(2))**2+(FORCE(I,J,2)*X(1))**
     12+(FORCE(I,J,4)*X(3))**2
C     TOOL LIFE CONSTRAINT
      CON(NCON)=-ALOG(COMD(I,J,3))+ALOG(10.)*TOOL(I,J,7)+(1.+TOOL(I,J,5)
     1)*X(2)+(1.+TOOL(I,J,1))*X(1)+TOOL(I,J,3)*X(3)-ALOG(X(4))+PHIP1*(SQRT(G1))
     1RT(G1))
      CON(NCON)=CON(NCON)+ALOG(COMD(I,J,3))-ALOG(COMD(I,J,8))
      NCON=NCON+1
C     MAXIMUM FORCE CONSTRAINT
      CON(NCON)=ALOG(TCM(I,J,8))-ALOG(10.)*FORCE(I,J,7)-FORCE(I,J,1)*X(2)
     1)-FORCE(I,J,5)*X(1)-FORCE(I,J,3)*X(3)+PHIP2*(SQRT(G2))
      FC=(10.**FORCE(I,J,7))*(EXP(X(2))**FORCE(I,J,5))*(EXP(X(1))**
     1FORCE(I,J,1))*(EXP(X(3))**FORCE(I,J,3))
      HP1(I,J)=ALOG(CHP)+ALOG(FC)+X(2)+ALOG(COMD(I,J,7)+COMD(I,J,8)/2.)
      HP2(I,J)=PHIP1*(HP1(I,J))*((FORCE(I,J,8))**2)+(X(1)*FORCE(I,J,2))**2
     1*2+(X(2)*FORCE(I,J,4))**2+(FORCE(I,J,6)*X(3))**2
      NCON=NCON+1
C     SURFACE ACCURACY CONSTRAINT
      IF(COMD(I,J,13).EQ.0.) CON(NCON)=-X(1)+ALOG(2.*COMD(I,J,16)
     1/(SIN(2.*TCM(I,J,5))))
        IF(COMD(I,J,13).EQ.1.) CON(NCON)=2.*X(1)+ALOG(COMD(I,J,16)*8.
     1*TCM(I,J,M))
      NCON=NCON+1
C     NO. OF COMPONENTS CONSTRAINT
      CON(NCON)=ALOG(COMR(1))-X(4)
      NCON=NCON+1
C     TOOL VIBRATION CONSTRAINT
      CON(NCON)=ALOG(TCM(I,J,3))-X(3)-X(1)
      NCON=NCON+1
C     HORCE POWER CONSTRAINT
        CON(NCON)=CMC(MN,11)*CMC(MN,15)-HP1(I,J)-HP2(I,J)
          NJ(1,I,J)=1
          NJ(2,I,J)=2
          NJ(3,I,J)=3
      NCON=NCON+1
      CON(NCON)=X(1)
      NCON=NCON+1
      CON(NCON)=X(2)
      NCON=NCON+1
        CON(NCON)=X(3)
      NCON=NCON+1
      CON(NCON)=X(4)
      N=4
```

```
      NIN=NCCN
      RETURN
      END




      SUBROUTINE CONSTR(X)
C*****************************************************************************
C*****************************************************************************
      DIMENSION X(2 ),Y(20),NSPO(10),AMATR(10),COMR(20),COMD(5,2,25),
     1CMC(2,2 ),TCM(5,2,25),COST(10),TIMEC(5,2,10),TOOL(5,2,15),FORCE(5,2,15)
     12,15),PROB(10),T(10,10),TC(10,10),NJ(3,5,5),CON(100),HP1(10,10),
     1HP2(10,10)
      COMMON/TOTAL/NCONST,NOBJ,NTFN
      COMMON/SERCH2/FUNVAL
      COMMON/CYC/CYCLT,AMAXO
      COMMON/PRD/PROB
       COMMON/INVAR/INXE,INYE
      COMMON/MATV/AMATR,COMR,COMD,CMC,TCM,COST,TIMEC,TOOL,FORCE,NSPO
      COMMON/NVAR/NAMAT,NCOMR,NSTN,NOPT,NCOMD,NCMC,NTCM,NCOST,NTIME,NTP,
     1NFP,NMACIN,NSPNDL,NAXIL,NCROSS,NPROB,MN,NSTION,NOP
      COMMON/IND/INDEX,INDEXF,INDEXM
       COMMON/OUT/Y,IOY,IOX,NJ,G1,G2,FC,HP1,HP2,T,TC
      COMMON/MANE/N,NIN
      COMMON/INEQLT/CON,SATSFY
      DIMENSION TLIF(10)
      DIMENSION TVARI(10),FVARI(10)
      COMMON/LIFE/TLIF,TVARI,FVARI,NTA
      COMMON/TOR/FC2(10),G11(10),G22(10)
       DIMENSION PROBHP(5)
      NTA=0
      IF(INDEX.EQ.0.AND.INDEXM.EQ.2) GO TO 25
      IF(INDEX.EQ.1.AND.INDEXM.EQ.1) GO TO 31
      IF(INDEX.EQ.0.AND.INDEXM.EQ.1) GO TO 400
      CALL INTAL(X,Y)
C     INITIALISATION OF X AND Y
      INDEX=0
      GO TO 27
 31   RETURN
 25    DO28I=1,IOY
      IDX=IOX+I
      Y(I)=X(IDX)
 28    CONTINUE
  27   IF(INDEXM.EQ.1) GO TO 400
      NCONST=NCONST+1
      NCON=0
      IOAXIL=0
      NSPNDL=1
 12   I=NSPNDL
      NN=NSPC(I)
      DO2J=1,NN
```

```
      IF(COMD(I,J,9).EQ.1.) GO TO 2
      IF(COMD(I,J,1).EQ.1.) GO TO 3
      IF(COMD(I,J,6).GT.1.) GO TO 4
      NTA=NTA+1
      IF(1.GT.1) COMD(I,J,7)=2.                                         CARE
      IF(I.NE.1.AND.COMD(I,J,1).EQ.1.)COMD(I,J,6)=2.                    CARE
      INX=I+1-IOAXIL
      PHIP1=PROB(1)
      PHIP2=PROB(2)
      CHP=TCM(I,J,12)*(10.**(-5))
      INX=1                                                             CARE
      NJ(1,I,J)=INX
      IOX=INX
      NCON=JCUN+1
C     MAXIMUM FEED CONSTRAINT
      CON(NCCN)=ALOG(CMC(MN,8))-X(INX)
      NCON=NCON+1
      CON(NCCN)=-ALOG(CMC(MN,7))+X(INX)
      GO TO 5
 4    INX=IOX
       NJ(1,I,J)=INX
      NTA=NTA+1
 5    IF(COMD(I,J,7).GT.1.) GO TO 6
      INY=I+1
      INY=2                                                             CARE
      NJ(2,I,J)=INY
      IOY=INY
      NCON=NCON+1
C     MAXIMUM SPEED CONSTRAINT
      CON(NCCN)=ALOG(CMC(MN,6))-Y(INY)
      NCON=NCON+1
C     MINIMUM SPEED CONSTRAINY
        CON(NCON)=-ALOG(CMC(MN,5))+Y(INY)
      GO TO 8
 6    INY=IOY
      NJ(2,I,J)=INY
      GO TO 8
 3    INX=1
      NTA=NTA+1
      IOAXIL=IOAXIL+1
      NCON=NCON+1
C     MAXIMUM FEED CONSTRAINT
      CON(NCCN)=ALOG(CMC(MN,20))-X(1)
       NJ(1,I,J)=1
      NCON=NCON+1
C     MINIMUM FEED CONSTRAINT
      CON(NCON)=-ALOG(CMC(MN,19))+X(1)
      IF(COMD(I,J,7).GT.1.) GO TO 8
      INY=I+1
      INY=2                                                             CA-E
      NJ(2,I,J)=INY
      IOY=INY
      NCON=NCON+1
```

```
C     MAXIMUM SPEED CONSTRAINT
      CON(NCON)=ALOG(CMC(MN,6))-Y(INY)
      NCON=NCON+1
C     MINIMUM SPEED CONSTRAINY
      CON(NCON)=-ALOG(CMC(MN,5))+Y(INY)
 8    NCON=NCON+1
      G1=TOOL(I,J,8)**2+(TOOL(I,J,6)*Y(INY))**2+(TOOL(I,J,2)*X(INX))**2+(TOOL(I,
     1(TOOL(I,J,4)*COMD(I,J,4))**2
      G1=(TOOL(I,J,8)*ALOG(10.))**2-TOOL(I,J,8)**2+G1
C     TOOL LIFE CONSTRAINT
      CON(NCON)=ALOG(10.)*TOOL(I,J,7)+(1.+TOOL(I,J,1))*X(INX)+(1.+TOOL(I
     1,J,5))*Y(INY)+TOOL(I,J,3)*COMD(I,J,4)-Y(1)+PHIP1*(SQRT(G1))
      CON(NCON)=CON(NCON)-ALOG(COMD(I,J,8))
      TLIFE=CON(NCON)-PHIP1*SQRT(G1)+Y(1)
      TLIF(NTA)=EXP(TLIFE)
      TVARI(NTA)=TLIF(NTA)*SQRT(G1)
      G2=FORCE(I,J,8)**2+(FORCE(I,J,6)*Y(INY))**2+(FORCE(I,J,2)*X(INX))**2
     1*1+(FORCE(I,J,4)*COMD(I,J,4))**2
      G2=G2-FORCE(I,J,8)**2+(FORCE(I,J,8)*ALOG(10.))**2
      NCON=NCON+1
C     MAXIMUM FORCE CONSTRAINT
      CON(NCON)=ALOG(TCM(I,J,8))-ALOG(10.)*FORCE(I,J,7)-FORCE(I,J,5)*Y(IN
     1NY)-FORCE(I,J,1)*X(INX)-FORCE(I,J,3)*COMD(I,J,4)+PHIP2*(SQRT(G2))
      FC=(10.**FORCE(I,J,7))*(EXP(Y(INY))**FORCE(I,J,5))*(EXP(X(INX))**
     1FORCE(I,J,1))*(EXP(COMD(I,J,4))**FORCE(I,J,3))
      FVARI(NTA)=FC*SQRT(G2)
      G22(NTA)=G2
      G11(NTA)=G1
      FC2(NTA)=FC
       HP1(I,J)=CHP*FC*EXP(Y (INY))*(COMD(I,J,19)+COMD(I,J,20))/2.
 2    CONTINUE
      NCON=NCON+1
C     SURFACE ACCURACY CONSTRAINT
      +F(COMD(+,J,13).EQ.0.) CON(NCON)=-X(1)+ALOG(2.*COMD(+,J,16)
     1/(SIN(2.*TCM(I,J,5))))
        IF(COMD(I,J,13).EQ.1.) CON(NCON)=2.*X(1)+ALOG(COMD(I,J,16)*8.
     1*TCM(I,J,M))
      NSPNDL=NSPNDL+1
      IF(NSPNDL.GT.NSTION) GO TO 11
      GO TO 12
 11   NCON=NCON+1
       PROBHP(1)=0.
      SHP=0.
      DO401I=1,NSTION
      NN=NSPC(I)
       DO40J=1,NN
      IF(COMD(I,J,9).EQ.0.) GO TO 40
      PROBHP(1)=PROBHP(1)+ALOG(10.)*HP1(I,J)
      PROBHP(1)=(PROBHP(1)*FORCE(I,J,8))**2
 40   CONTINUE
 401   CONTINUE
      SHP=SHP+PROBHP(1)*PHIP2
      PROBHP(2)=0.
```

```
      DO411I=1,NSTION
      NN=NSPC(I)
      DO41J=1,NN
       IF(COMD(I,J,9).EQ.0.) GO TO 41
      K=NJ(2,I,J)
      PROBHP(2)=PROBHP(2)+Y(K)*HP1(I,J)
      PROBHP(2)=(PROBHP(2)*FCRCE(I,J,6))**2
 41    CONTINUE
 411   CONTINUE
       SHP=SHP+PROBHP(2)*PHIP2
      PROBHP(3)=0.
      DO422I=1,NSTION
      NN=NSPC(I)
      DO42J=1,NN
       IF(COMD(I,J,9).EQ.0.) GO TO 42
       K=NJ(1,I,J)
      PROBHP(3)=PROBHP(3)+X(K)*HP1(I,J)
      PROBHP(3)=(PROBHP(3)*FCRCE(I,J,2))**2
 42   CONTINUE
 422   CONTINUE
      SHP=SHP+PROBHP(3)*PHIP2
      PROBHP(4)=0.
      DO433I=1,NSTION
      NN=NSPO(I)
      DO43J=1,NN
       IF(COMD(I,J,9).EQ.0.) GO TO 43
       PROBHP(4)=PROBHP(4)+ COMD(I,J,4)*HP1(I,J)
      PROBHP(4)=(PROBHP(4)*FCRCE(I,J,4))**2
 43   CONTINUE
 433   CONTINUE
      SHP=SHP+PROBHP(4)*PHIP2
      SHP=SQRT(SHP)*SQRT(PHIP2)
      CON(NCCN)=CMC(MN,11)*CMC(MN,15)+SHP
      DO9I=1,NSTION
      N=NSPO(I)
      DO10J=1,N
      IF(COMD(I,J,9).EQ.0.) GO TO 10
C     HORCE POWER CONSTRAINT
       CON(NCON)=CON(NCON)-HP1(I,J)
 10    CONTINUE
 9    CONTINUE
      DO26I=1,IOY
      IDX=IOX+I
      X(IDX)=Y(I)
 26   CONTINUE
       FUNVAL=FUNXON(X)
      DO45I=1,NSTION
      NN=NSPC(I)
      DO46J=1,NN
       IF(COMD(I,J,9).EQ.0.) GO TO 46
      NCON=NCON+1
       CON(NCON)=CYCLT-T(I,J)+.01
 46   CONTINUE
```

```
 45    CONTINUE
C     NO. OF COMPONENTS CONSTRAINT
      NCON=NCON+1
      CON(NCON)=ALOG(COMR(1))-Y(1)
      NIN=NCON
      N=IDX+IDY
      RETURN
      END



      SUBROUTINE MINIMA(X)
C     ********************************************************************
C     *   THIS SUBROUTINE IS THE BACKBONE OF THE PROGRAMME  . IT CALLS VARIOUS  *
C     *   SUBROUTINES AND FUN.SUBPROGRAMME DIRECTLY/INDIRECTLY              *
C     ***************************************************************M************
      COMMON/DIV/C1
      COMMON/MANE/N,NIN
      COMMON/MINI/R
      COMMON/POVEL/Z
      COMMON/SERCH2/FUNVAL
      COMMON/INEQLT/CON,SATSFY
      DIMENSION CON(100),X(50),X1(100)
C     ********************************************************************
C     *   THE INITIAL VALUE OF *R* IN PENALTY FUNCTION PROGRAMME IS TAKEN 10    *
C     ********************************************************************
      I=0
      CALL FUNT(X,Z)
      IF(I.NE.0) GOTO 53
      PRINT 100,FUNVAL
      CALL OUTPUT(X)
      CALL CHECK(X)
      IF(SATSFY.EQ.1.) GOTO 53
      PRINT 59
 59   FORMAT(//20X,*INITIAL SOLUTION DOES NOT LIE IN CONSTRAINT SET*)
      RETURN
 53   CALL POWELL(X)
      I=I+1
      IF(I.EQ.1) GOTO 80
      DIFF=ABS(Z-Z1)
C     ********************************************************************
C     *    THE TERMINATION CRITERIA                                          *
C     *   TERMINATE WHEN DIFF.BETWEEN PRIMARY AND SECONDARY FUNCTION BECOMES    *
C     *   LESS THAN 10**-6 TIMES THE PRIMARY FUNCTIONAL VALUE               *
C     ********************************************************************
      IF(DIFF.LE.ABS(Z*1.E-03)) GO TO 90
 80   Z1=Z
C     ********************************************************************
C     *   PENALTY CONSTANT *R* REDUCED TO 1/4 OF ITS ORIGINAL VALUE         *
C     ********************************************************************
      R=R/C1
      GO TO 53
```

```
100   FORMAT(///,10X,*INITIAL FUNCTION VALUE*,F15.6,50X,*SOLUTION VECTO
     1R *)
 90   R=0.
      CALL POWELL(X)
      PRINT75
 75   FORMAT(////////,40X,* THE OPTIMUM SOLUTION HAS REACHED *,/,1H1)
      PRINT 2,(CON(I),I=1,NIN)
 2    FORMAT(1X,10F12.4)
      PRINT 50 ,FUNVAL
 50   FORMAT(20X,*OPTIMAL VALUE OF OBJECTIVE FUNCTION=*,F10.4)
      CALL OUTPUT(X)
      RETURN
      END


      SUBROUTINE GRAD(X,G)
C     **********************************************************************
C     *THIS SUBROUTINE EVALUATES THE GRADIENT BY FORWARD DIFF. FORMULAE    *
C     **********************************************************************
      COMMON/MANE/N,NIN
      COMMON/INEQLT/CON,SATSFY
      COMMON/POVEL/Z
      DIMENSION X(50),X1(50),G(50)
      DIMENSION CON(100)
      FX=Z
      DO 100 I=1,N
      DO 10 J=1,N
   10 X1(J)=X(J)
      DX=X(I)*0.1
      IF(DX.EQ.0.) DX=0.01
    5 X1(I)=X(I)+DX
      CALLCHECK(X1)
      IF(SATSFY.EQ.1.) GOTO20
      IF(DX.LT.0.) GOTO11
      DX=-DX
      GOTO 5
 11   DX=-DX*0.1
      GOTO5
 20   CALL FUNT(X1,FX1)
  100 G(I)=(FX1-FX)/DX
      RETURN
      END

      FUNCTION FUNXON(X)
C     **********************************************************************
C     *    THIS FUNCTION SUB PROGRAMM EVALUATES THE OBJECTIVE FUNCTION     *
C     **********************************************************************
      COMMON/TOTAL/NCONST,NOBJ,NTFN
      COMMON/CYC/CYCLT,AMAXO
      COMMON/MANE/N,NIN
      COMMON/MATV/AMATR,COMR,COMD,CMC,TCM,COST,TIMEC,TOOL,FORCE,NSPO
      COMMON/NVAR/NAMAT,NCOMR,NSTN,NOPT,NCOMD,NCMC,NTCM,NCOST,NTIME,NTP,
```

```
1NFP,NMACIN,NSPNOL,NAXIL,NCROSS,NPROB,MN,NSTION,NOP
 COMMON/IND/INDEX,INDEXF,INDEXM
 COMMON/OUT/Y,IOY,IOX,NJ,G1,G2,FC,HP1,HP2,T,TC
 COMMON/INEQLT/CON,SATSFY
 DIMENSION NSPC(10),AMATR(10),COMR(10),COMD(8,6,25),CMC(5,20),
1TCM(8,6,25),COST(10),TIMEC(8,6,10),TCCL(8,6,15),FORCE(8,6,15),
2PROB(10),CON(   ),Y(5  ),HP1(10,10),HP2(10,10),X(50),T(10,10),TC(10,10)
3,10),NJ(3,8,5)
  DIMENSION TMAT(50),TMATC(50)
 IF(INDEXM.EQ.1) RETURN
 NOBJ=NOBJ+1
 NTFN=NTFN+1
 CYCLT1=.25
 KN=
 COST(10)=  .
 DO1I=1,NSTION
 NN=NSPC(I)
 DO2J=1,NN
 IF(COMD(I,J,9).EQ.0.) GO TO 2
 INX=NJ(1,I,J)
 INY=NJ(2,I,J)
 T(I,J)=COMD(I,J,3)/(EXP(X(INX))*EXP(Y(INY)))
 COST(10)=COST(10)+T(I,J)*COST(1)
 TC(I,J)=ABS(T(I,J)-CYCLT1)
 KN=KN+1
 TMAT(KN)=T(I,J)
    TMATC(KN)=TC(I,J)
2    CONTINUE
1    CONTINUE
     CALL MAXM(TMAT,CYCLT,J,KN)
     GO TO(600,700,800),INDEXF
600   FUNXON=CYCLT
     RETURN
700   CALL CBJFN(TCOST)
     FUNXON=TCOST
     RETURN
800   CALL CBJFN(TCOST)
     FUNXON=TCOST
     RETURN
     END


     SUBROUTINE POWELL(X)
C    ***********************************************************************
C     *    THIS SUBROUTINE PERFORMS D-F-P OPERATIONS                       *
C    ***********************************************************************
     COMMON/MINI/R
     COMMON/POVEL/Z
      COMMON/SERCH2/FUNVAL
     COMMON/SERCH3/F
     COMMON/MANE/N,NIN
     COMMON/INEQLT/CON,SATSFY
     DIMENSION X(50),G(50),H(20,20),S(20),SIGMA(20),YY(20),H1(20,20),
```

```
     1G1(50),PO(50),EPSLON(50)
      DIMENSION CON(100)
      NM1=N-1
      DO 75 I=1,N
C    *    THE EPSLON GIVES THE ACCURACY OF STEP INCREMENT
 75   EPSLON(I)=10.**(-5)
 71   DO 80 I=1,N
      DO 80 J=1,N
      H(I,J)=0.
 80   H(I,I)=1.
      CALL GRAD(X,G)
      ITN=1
 90   DOTSS=0.
      DOTSIG=0.
      D=0.
      DO 91 I=1,N
      S(I)=0.
      DO 910 J=1,N
 910  S(I)=S(I)-H(I,J)*G(J)
      D=D+G(I)*S(I)
C
C
      IF(D.GE.0.) GO TO 71
 91   DOTSS=S(I)*S(I)+DOTSS
      DO 990 I=1,N
      S(I)=S(I)/SQRT(DOTSS)
 990  CONTINUE
      CALL SEARCH(X,S,ALPI)
C
C      CONVERGENCE CRITERIA STEP SIZE
C
      IF(ABS(ALPI).LT.0.1E-02) RETURN
C   VALUE OF ALPHA=0 IN LINEAR MIN. MEANS NO PROGRESS IN LINEAR MIN.
      IF(ITN.EQ.1) GOTO 920
      IF(Z.LT.Z1) GOTO 920
C   THIS FOLLOWS F(X0+ALPHA*S) = FO
      Z=Z1
      RETURN
 920  DO 92 I=1,N
      SIGMA(I)=ALPI*S(I)
      ET=AMAX1(1.,ABS(ALPI))
      DO 76 L=1,N
C
C     CONVERGENCE FOR TOTAL MOVE
C
      IF(ABS(ET*S(L)).LE.EPSLON(L)) GOTO 76
      GOTO 89
 76   CONTINUE
      PRINT 5000,R,ITN,Z,F,ALPI,(S(J),J=1,N)
      CALL OUTPUT(X)
      RETURN
 89   DOTSIG=DOTSIG+SIGMA(I)**2
 92   X(I)=X(I)+SIGMA(I)
```

```
      CALL GRAD(X,G1)
      DR=0.
      AR1=0.
      DO 93 I=1,N
      DR=DR+G1(I)*S(I)
      YY(I)=G1(I)-G(I)
      AR1=G(I)*S(I)+AR1
 93   G(I)=G1(I)
C
C     THE FOLLOWING STATEMENT IS TO SKIP THE UPDATING OF H-MATRIX IF REQUIRED
C
      IF(ABS(DR/AR1).GT.0.3E-01) GO TO 96
      A1=0.
      A2=0.
      DO 95 I=1,N
      A1=A1+SIGMA(I)*YY(I)
      DO 95 J=1,N
 95   A2=A2+YY(I)*H(I,J)*YY(J)
      A1=1./A1
      A2=-1./A2
      DO 98 I=1,N
      DO 98 J=1,N
      H1(I,J)=H(I,J)+SIGMA(I)*SIGMA(J)*A1
      DO 98 K=1,N
      DO 98 M=1,N
 98   H1(I,J)=H1(I,J)+A2*H(I,K)*YY(K)*YY(M)*H(M,J)
      DO 99 I=1,N
      DO 99J=1,N
 99   H(I,J)=H1(I,J)
 86   ITN=ITN+1
      PRINT 5000,R,ITN,Z,I,ALPHA,(S(I),I=1,N)
      CALL OUTPUT(X)
      IF(ITN.EQ.5*N)  GO TO 106
C     THUS 20 IS THE MAXM. NO. OF ITERATIONS FOR A VALUE OF *R*
 100  Z1=Z
      GOTO 90
 106  PRINT 107
 107  FORMAT(///10X,*MAXIMUM NO OF ITERATIONS EXCEEDED HENCE QUIT*)
 5000 FORMAT(1X,*R=*,F12.6,I5,3F11.4,11F7.3)
 105  RETURN
      END


      SUBROUTINE INIT(X)
C
C     THIS SUBROUTINE TO INITIALIZE THE VARIABLES  IN CASE O F SINGLE- TOOL CASE
C
      DIMENSION X(50)
      X(1)=200.
       X(2)=.02
      X(3)=330.
       X(4)=.2
      RETURN
```

```
      SUBRCUTINE FUNCT(X,FX)
C     *******************************************************************
C     *    THIS SUBRCUTINE EVALUATES THE SECCMDARY FUNCTICN
C     *******************************************************************
      COMMON/MINI/R
      COMMON/MANE/N,NIN
      COMMON/INEQLT/CON,SATSFY
      COMMON/SERCH2/FUNVAL
      DIMENSION CON(100)
       DIMENSION X(50)
      FX=FUNVAL
      DO 10 I=1,NIN
   10 FX=FX+R/CON(I)
      RETURN
      END

      SUBRCUTINE CHECK(X)
C     *******************************************************************
C     *    THIS SUBRCUTINR IS TO DHECK THE FEASIBILITY REGION
C     *******************************************************************
      COMMON/MANE/N,NIN
      COMMON/INEQLT/CON,SATSFY
      DIMENSION CON(100)
      DIMENSION X(50)
      SATSFY=0.
      CALL CONSTR(X)
      DO 10 I=1,NIN
      IF(CON(I).LE.0.) GOTO 15
   10 CONTINUE
      RETURN
   15 SATSFY=1.
      RETURN
      END

      SUBRCUTINE SEARCH(X,S,EMID)
C     *******************************************************************
C     *    THIS SUBRCUTINE PERFORMS THE GCLDEN SEARCH OPERATIONS
C     *******************************************************************
      COMMON/SERCH2/FUNVAL
      COMMON/POVEL/Z
      COMMON/SERCH3/F
      COMMON/MANE/N,NIN
      COMMON/INEQLT/CON,SATSFY
      COMMCN/STEP/ETA
      DIMENSION CON(100)
        DIMENSION X(50),S(50)
      DIMENSION YY(50)
      CONST1=0.381966
```

```
      CONST2=[illegible]
      ICD=[illegible]
      J=[illegible]
100   DO 110 I=[illegible]
110    YY(I)=X(I)+ETA*S(I)
      CALL CHECK(YY)
      IF(SATSFY.EQ.[illegible]) GOTO 93
      J=[illegible]
      ETA=ETA/[illegible]
      GOTO 100
 93   IF(J.NE.[illegible]) GOTO 94
      ETA=2.*ETA
      GOTO [illegible]
 94   EMIN=ETA
      EMAX=2.*ETA
113     DO 111 I=[illegible],N
111    YY(I)=X(I)+EMAX*S(I)
       CALL CHECK(YY)
      IF(SATSFY.EQ.[illegible]) GO TO 112
        EMAX=.9*EMAX
      GO TO 113
112   EMIN1=EMIN
      EMAX1=EMAX
 10   DIST=EMAX-EMIN
      DE=DIST
 11   E1=EMIN+CONST1*DE
      E2=EMIN+CONST2*DE
      CALL FUN1(X,S,E1,F1)
      FU1=FUNVAL
      CALL FUN1(X,S,E2,F2)
      FU2=FUNVAL
 12   IF(ABS(EMAX-EMIN).LE.0.001*ABS(DIST)) GOTO 30
      DE=CONST2*DE
      IF(F1-F2) 13,25,14
 14   EMIN=E1
      E1=E2
      F1=F2
      FU1=FU2
      E2=EMIN+CONST2*DE
      CALL FUN1(X,S,E2,F2)
      FU2=FUNVAL
      GOTO 12
 13   EMAX=E2
      F2=F1
      E2=E1
      FU2=FU1
      E1=EMIN+CONST1*DE
      CALL FUN1(X,S,E1,F1)
      FU1=FUNVAL
      GOTO 12
25     IF(ICD.EQ.1) GO TO 13
      EMIN=E1
      EMAX=E2
```

```
      DE=EMAX-EMIN
      GOTO [illegible]
 30   IF(F1.GT.F2) GOTO 32
      EMID=E1
      Z=F1
      F=FU1
      ICD=ICD+1
      IF(ICD.GT.2) GO TO 33
       EMIN=0.
      EMAX=EMID
       GO TO 112
 32   EMID=E2
      Z=F2
      F=FU2
      ICD=ICD+1
      IF(ICD.GT.2) GO TO 33
       EMIN=0.
      EMAX=EMID
       GO TO 112
33    RETURN
      END


      SUBROUTINE FUN1(X,S,E,F)
      COMMON/MANE/N,NIN
      COMMON/INEQLT/CON,SATSFY
      DIMENSION X(50),S(50),YY(50)
       DIMENSION CON([illegible])
      DO 10 I=1,N
10    YY(I)=X(I)+E*S(I)
      CALL CONSTR(YY)
      CALL FUNT(YY,F)
      RETURN
      END

      SUBROUTINE MAXM(E,AMAX,J,N)
      DIMENSION E(50)
      AMAX=-10.**4
      DO10I=1,N
      IF(E(I).LE.AMAX) GO TO 10
      AMAX=E(I)
      J=I
10    CONTINUE
      RETURN
      END
```